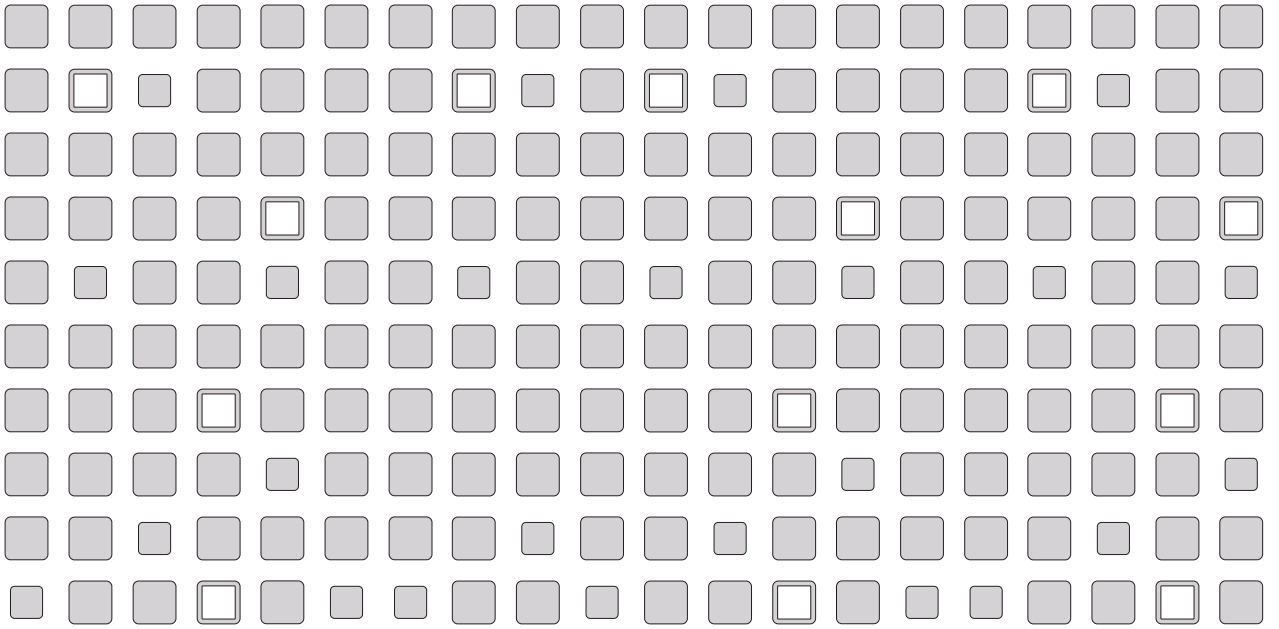
GSX Server™ 3

Enterprise-Class Virtual Infrastructure for Intel-Based Servers

Virtual Machine Guide



vmware®

VMware, Inc.

3145 Porter Drive
Palo Alto, CA 94304
www.vmware.com

Please note that you will always find the most up-to-date technical documentation on our Web site at http://www.vmware.com/support/.

The VMware Web site also provides the latest product updates.

Copyright © 1998-2005 VMware, Inc. All rights reserved. Protected by one or more of U.S. Patent Nos. 6,397,242, 6,496,847, 6,704,925, 6,711,672, 6,725,289, 6,735,601, 6,785,886, 6,789,156 and 6,795,966; patents pending. VMware, the VMware "boxes" logo and design, Virtual SMP and VMotion are registered trademarks or trademarks of VMware, Inc. in the United States and/or other jurisdictions. Microsoft, Windows and Windows NT are registered trademarks of Microsoft Corporation. Linux is a registered trademark of Linus Torvalds. All other marks and names mentioned herein may be trademarks of their respective companies. Revision 20050707 Version: 3.2 Item: GSX-ENG-Q205-090

Table of Contents

Introduction and System Requirements **11**
VMware GSX Server: Enterprise-Class Virtual Infrastructure for x86-Based Servers 12
Welcome to VMware GSX Server 14
Enterprise-Class Virtual Infrastructure for Intel-Based Servers 15
What's New in Version 3 16
Features in Version 3.2 16
Critical Bugs Fixed in GSX Server 3.2 17
Features in Version 3.1 17
Critical Bugs Fixed in GSX Server 3.1 18
Features in Version 3 19
Virtual Machine Specifications 24
Supported Guest Operating Systems 27
Technical Support Resources 29
Self-Service Support 29
Online and Telephone Support 29
Support Offerings 29
Reporting Problems 29
Log Files 31

Creating a New Virtual Machine **35**
Setting Up a New Virtual Machine 36
What's in a Virtual Machine? 36
Creating a New Virtual Machine with the New Virtual Machine Wizard 38
Creating a New Virtual Machine from the VMware Management Interface 50
Installing a Guest Operating System 56
Example: Installing Windows Server 2003 as a Guest Operating System 56

Using VMware Tools **59**
About VMware Tools 60
Installing VMware Tools 62
Installing VMware Tools in a Windows Virtual Machine 62
Installing VMware Tools in a Linux or FreeBSD Virtual Machine 71

Installing VMware Tools in a NetWare Virtual Machine 74
Executing Scripts When the Virtual Machine's Power State Changes 76
Issues to Consider 77
Configuring VMware Tools 78
Configuring VMware Tools in a Windows Virtual Machine 78
Configuring VMware Tools in a Linux or FreeBSD Virtual Machine 82
Configuring VMware Tools in a NetWare Virtual Machine 85
About the VMware Tools Service 90
Synchronizing the Time in the Guest with the Host Operating System 91
Executing Commands After You Power Off or Reset a Virtual Machine 92
Passing a String from the Host Operating System to the Guest Operating System 92
Passing Information Between the Guest Operating System and a VMware Scripting API Script 94

Running Virtual Machines 95
Overview of the VMware Virtual Machine Console Window 97
Using the Home Tab 98
Using Tabs 102
Configuring a Virtual Machine 103
Using the Virtual Machine Inventory 103
Displaying Hints 104
Checking the Status of VMware Tools 104
Creating a Screen Shot of a Virtual Machine 105
Connecting to Virtual Machines and GSX Server Hosts 106
Connecting to a Virtual Machine from a Windows Host or Client 107
Connecting to a Virtual Machine from a Linux Host or Client 110
Connecting to a Virtual Machine from the VMware Management Interface 111
Connecting to a Different GSX Server Host 112
Connecting to Older GSX Server and ESX Server Systems and Older Virtual Machines 113
Changing the Power State of a Virtual Machine 118
Using Power Options for Virtual Machines 118
Suspending and Resuming Virtual Machines 121
Shutting Down a Virtual Machine 121

Powering Virtual Machines On and Off When the Host Starts and Shuts Down ____ 122
Controlling the Virtual Machine Display ____ 125
Using Full Screen Mode ____ 125
Using Quick Switch Mode ____ 125
Taking Advantage of Multiple Monitors ____ 126
Fitting the Console Window to the Virtual Machine ____ 127
Fitting a Windows Guest Operating System's Display to the VMware Virtual Machine Console Window ____ 127
Simplifying the Screen Display ____ 128
Taking and Reverting to a Snapshot ____ 129
Running Virtual Machines from DVD-ROM or CD-ROM Discs ____ 130
Using PXE with Virtual Machines ____ 132
Sharing Files Between Guest and Host Operating Systems ____ 134
Sharing Files Between Two Windows Systems ____ 135
Sharing Files by Connecting to a Windows System from a Linux System ____ 135
Sharing Files by Connecting to a Linux System from a Windows System ____ 136
Sharing Files Between Two Linux Systems ____ 138
Installing Software in a Virtual Machine ____ 139
Cutting, Copying and Pasting Text ____ 140
Using Devices in a Virtual Machine ____ 141
Adding, Configuring and Removing Devices in a Virtual Machine ____ 141
Connecting and Disconnecting Removable Devices ____ 144
Command Reference ____ 145
Startup Options on a Linux Host ____ 145
Startup Options on a Windows Host ____ 146
Using Keyboard Shortcuts ____ 147

Preserving the State of a Virtual Machine ____ 149
Suspending and Resuming Virtual Machines ____ 150
Setting the Suspended State File Directory ____ 151
Taking Snapshots ____ 153
What Is Captured by a Snapshot? ____ 154
Ways of Using Snapshots ____ 155
Snapshots and a Virtual Machine's Hard Disks ____ 156
Snapshots and Other Activity in the Virtual Machine ____ 156

Settings for Snapshots 157
Removing the Snapshot 157
Snapshots and Legacy Disk Modes 158
Snapshots and Repeatable Resume 158
Snapshots and Legacy Virtual Machines 158

Using Disks in a Virtual Machine 161
Configuring Hard Disk Storage in a Virtual Machine 162
Disk Types: Virtual and Physical 162
Additional Information about Disk, Redo-Log, Snapshot and Lock Files 165
Defragmenting and Shrinking Virtual Disks 168
Configuring Optical and Floppy Drives 171
Configuring Virtual DVD-ROM and CD-ROM Drives 171
Configuring Virtual Floppy Drives 175
Adding Drives to a Virtual Machine 177
Adding Virtual Disks to a Virtual Machine 177
Adding Physical Disks to a Virtual Machine 184
Adding DVD-ROM or CD-ROM Drives to a Virtual Machine 187
Adding Floppy Drives to a Virtual Machine 189
Using VMware Virtual Disk Manager 191
Running the VMware Virtual Disk Manager Utility 192
Shrinking Virtual Disks with VMware Virtual Disk Manager 195
Examples Using the VMware Virtual Disk Manager 195
Configuring a Dual-Boot Computer for Use with a Virtual Machine 198
Configuring Dual- or Multiple-Boot Systems to Run with GSX Server 200
Setting Up Hardware Profiles in Virtual Machines 206
Running a Windows 2000, Windows XP or Windows Server 2003 Virtual Machine from an Existing Multiple-Boot Installation 210
Setting Up the SVGA Video Driver for a Windows 95 Guest Operating System Booted from a Physical Disk 210
Setting Up the SVGA Video Driver for Use with a Windows 98 Guest Operating System Booted from a Physical Disk 211
Do Not Use Windows 2000, Windows XP and Windows Server 2003 Dynamic Disks as Physical Disks 213
Configuring Dual- or Multiple-Boot SCSI Systems to Run with VMware GSX Server on a Linux Host 214

Installing an Operating System onto a Physical Partition from a Virtual Machine ____ 220
Configuring a Windows Host ____ 221
Configuring a Linux Host ____ 223
Disk Performance in Windows NT Guests on Multiprocessor Hosts ____ 225
Improving Performance ____ 225

Networking ____ 227
Components of the Virtual Network ____ 229
Common Networking Configurations ____ 231
Bridged Networking ____ 231
Network Address Translation (NAT) ____ 232
Host-Only Networking ____ 234
Custom Networking Configurations ____ 235
Changing the Networking Configuration ____ 238
Adding and Modifying Virtual Network Adapters ____ 238
Configuring Bridged Networking Options on a Windows Host ____ 242
Enabling, Disabling, Adding and Removing Host Virtual Adapters ____ 247
Advanced Networking Topics ____ 250
Selecting IP Addresses on a Host-Only Network or NAT Configuration ____ 250
Avoiding IP Packet Leakage in a Host-Only Network ____ 253
Maintaining and Changing the MAC Address of a Virtual Machine ____ 255
Controlling Routing Information for a Host-Only Network on a Linux Host ____ 256
Other Potential Issues with Host-Only Networking on a Linux Host ____ 257
Setting Up a Second Bridged Network Interface on a Linux Host ____ 258
Configuring Bridged Networking When Using Teamed Network Interface Cards on Your Host ____ 259
Setting Up Two Separate Host-Only Networks ____ 261
Routing Between Two Host-Only Networks ____ 264
Using Virtual Ethernet Adapters in Promiscuous Mode on a Linux Host ____ 267
Understanding NAT ____ 269
Using NAT ____ 269

The Host Computer and the NAT Network 270
DHCP on the NAT Network 270
DNS on the NAT Network 270
External Access from the NAT Network 271
Advanced NAT Configuration 272
Custom NAT and DHCP Configuration on a Windows Host 275
Considerations for Using NAT 276
Using NAT with NetLogon 277
Sample Linux vmnetnat.conf File 278
Using Samba for File Sharing on a Linux Host 280

Configuring Devices 289
Using Parallel Ports 290
About Parallel Ports 290
Adding a Parallel Port in a Virtual Machine 290
Configuring a Parallel Port on a Linux Host 292
Special Notes for the Iomega Zip Drive 295
Using Serial Ports 296
Using a Serial Port on the Host Computer 296
Using a File on the Host Computer 298
Connecting an Application on the Host to a Virtual Machine 300
Connecting Two Virtual Machines 303
Special Configuration Options for Advanced Users 306
Examples: Debugging over a Virtual Serial Port 307
Keyboard Mapping on a Linux Host 310
Quick Answers 310
The Longer Story 310
V-Scan Code Table 314
Using USB Devices in a Virtual Machine 319
Notes on USB Support 319
Adding a USB Controller 320
Connecting USB Devices 321
Using USB with a Windows Host 321
Replacing USB 2.0 Drivers on a Windows 2000 Host 322
Using USB with a Linux Host 322
Who Has Control over a USB Device? 322
Disconnecting USB Devices from a Virtual Machine 324
Human Interface Devices 324

Connecting to a Generic SCSI Device _____ 325
Device Support in Guest Operating Systems _____ 325
Adding a Generic SCSI Device to a Virtual Machine _____ 326
Generic SCSI on a Windows Host Operating System _____ 328
Generic SCSI on a Linux Host Operating System _____ 330

Video and Sound _____ 333
Setting Screen Color Depth in a Virtual Machine _____ 334
Changing Screen Color Depth on the Host _____ 334
Changing Screen Color Depth in the Virtual Machine _____ 335
Using Full Screen Mode on a Linux Host _____ 336
Configuring Sound _____ 337
Installing Sound Drivers in a Windows Server 2003 Guest Operating System _____ 337
Installing Sound Drivers in Windows 9x and Windows NT Guest Operating Systems _____ 337

Performance Tuning for Virtual Machines _____ 339
Allocating Memory to a Virtual Machine _____ 340
Configuring Virtual Machine Memory from a Console _____ 341
Configuring Virtual Machine Memory from the Management Interface _____ 342
Improving Performance for Guest Operating Systems _____ 343
Windows 2000, Windows XP and Windows Server 2003 Guest Operating System Performance Tips _____ 343
Windows 95 and Windows 98 Guest Operating System Performance Tips _____ 344
Linux Guest Operating System Performance Tips _____ 346

Glossary _____ 347

Index _____ 353

CHAPTER 1

Introduction and System Requirements

The following sections introduce you to VMware GSX Server.

- VMware GSX Server: Enterprise-Class Virtual Infrastructure for x86-Based Servers on page 12
- Welcome to VMware GSX Server on page 14
- What's New in Version 3 on page 16
- Virtual Machine Specifications on page 24
- Supported Guest Operating Systems on page 27
- Technical Support Resources on page 29

VMware GSX Server: Enterprise-Class Virtual Infrastructure for x86-Based Servers

VMware® GSX Server is virtual infrastructure for enterprise IT administrators who want to consolidate servers and streamline development and testing operations. GSX Server is easily installed and managed, and provides rapid return on investment (ROI). Unlike other virtualization products, GSX Server is enterprise-proven, preserves freedom of choice and offers an upgrade path to datacenter-class virtualization.

Because it has over four years of proven success, thousands of customers trust VMware GSX Server as their virtualization solution. Easily installed on Windows or Linux platforms, GSX Server provides advanced capabilities that make it the most flexible server virtualization product on the market. GSX Server is part of the widely deployed VMware virtual infrastructure solution with virtual machines compatible across all VMware products, and with unified management and provisioning provided by VMware VirtualCenter.

VMware GSX Server simplifies computing infrastructure by partitioning and isolating servers in secure and transportable virtual machines, each of which can run standard Windows, Linux or NetWare operating systems and applications. GSX Server allows you to remotely manage, automatically provision and standardize virtual machines on a secure, uniform platform.

Thousands of enterprise customers rely on VMware GSX Server to deliver server scalability, reliability and high availability and to maximize return on IT investments. Use GSX Server across the enterprise to:

- Streamline software development and testing operations with easily provisioned and managed server-based virtual machines.
- Implement server consolidation for new and legacy departmental server applications.
- Provision servers rapidly to local or remote locations.
- Streamline operating system and application patch management.

Key GSX Server 3 Benefits

GSX Server 3 offers the following key benefits:

- Shipping for over four years, with thousands of successful customers, VMware GSX Server is the most flexible and easily deployed server virtualization product on the market.
- Offers widest selection of supported host and guest operating systems of any virtualization technology — preserves your freedom to choose the best operating system platform.
- Integrates easily into any environment for ultimate versatility — installs like an application and runs on any standard x86 hardware.
- Supports large server — up to 64GB of host memory, 32 host processors and 64 powered-on virtual machines — for ultimate scalability, extensibility and robustness.
- Offers virtual machine compatibility across the entire VMware family of virtualization products and is ready to run with VMware VirtualCenter.
- Supports advanced virtual machine clustering for high-availability applications.
- Ensures reliable server consolidation through isolated virtual partitions.
- Allows secure remote management.
- Permits full network connectivity for virtual machines.
- Automates monitoring and control of virtual machines with VmCOM and VmPerl Scripting APIs.
- Runs virtual disk files on any system with current VMware software installed.

To discover more new features of VMware GSX Server 3, see What's New in Version 3 on page 16.

Welcome to VMware GSX Server

Thank you for choosing VMware GSX Server, the software that provides IT professionals with enterprise-class server consolidation and high availability of server resources by letting them run multiple operating systems in secure, transportable, high-performance virtual computers.

If you're new to GSX Server, this chapter is the place to start.

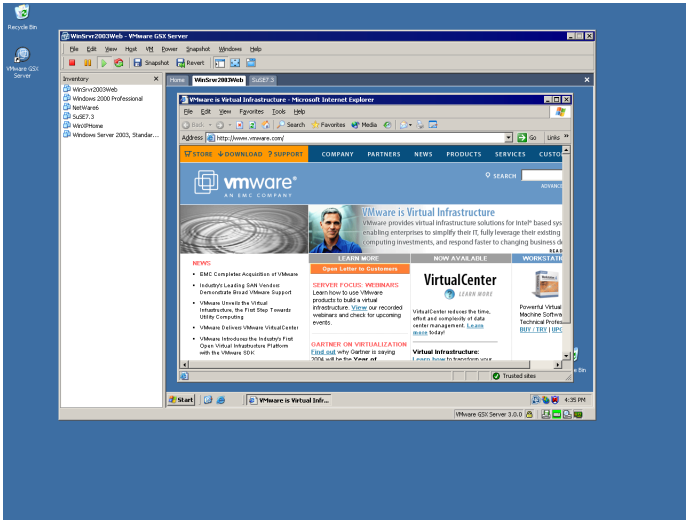
If you're a veteran user of VMware products, take a few minutes to see what's new in version 3 and review the notes on upgrading your installation.

The *VMware GSX Server Administration Guide* introduces you to some of the things you can do with GSX Server and guides you through the key steps for installing the software, configuring your server host and managing your virtual machines.

The *VMware GSX Server Virtual Machine Guide* provides information on creating virtual machines, as well as in-depth reference material for getting the most out of the sophisticated features of GSX Server.

Enterprise-Class Virtual Infrastructure for Intel-Based Servers

VMware products provide a virtualization layer that turns your physical computers into a pool of logical computing resources. You can then dynamically allocate those resources to any operating system or application in the way that best meets your needs. You'll be able to spend more time delivering tangible value to your business and less time installing operating systems, rebooting and reconfiguring hardware.



Run the operating systems you need — all at once.

With VMware virtualization technology, you can set up completely independent installations of operating systems on a single machine. Multiple instances of Microsoft® Windows® or Linux® can run side by side in virtual machines that you create with the GSX Server software. Each virtual machine is equivalent to a server with a unique network address and a full complement of hardware devices. You install and run a complete, unmodified operating system and application software, just as you do on a physical server.

Host and Guest

- The physical computer on which you install the GSX Server software is called the host computer, and its operating system is called the host operating system.
- The operating system running inside a virtual machine is called a guest operating system.
- For definitions of these and other special terms, see the glossary at the end of this manual.

What's New in Version 3

Whether you're a long-time power user of VMware GSX Server or a new user who is just learning what you can do with virtual machines, the new features in VMware GSX Server 3 extend its capabilities and make it easier to use.

Features in Version 3.2

Here are some highlights of key features to explore in VMware GSX Server 3.2:

New Operating System Support

VMware GSX Server 3.2 adds full support for the following 64-bit host operating systems on AMD64 and Intel® EM64T processors:

- Microsoft® Windows Server 2003 x64 Edition as a host operating system on AMD64 and Intel EM64T processors

VMware GSX Server 3.2 adds experimental support for the following 64-bit host operating systems on AMD64 and Intel EM64T processors:

- Red Hat™ Enterprise Linux 4
- Red Hat Enterprise Linux 3 Update 4
- SUSE™ LINUX Enterprise Server 9 Service Pack 1
- SUSE LINUX 9.3
- SUSE LINUX 9.2

VMware GSX Server 3.2 adds support for the following 32-bit host and guest operating systems:

- Microsoft Windows Server 2003 Service Pack 1
- Mandrake Linux 10.0 and 10.1
- Red Hat Enterprise Linux 4
- Red Hat Enterprise Linux 3 Update 4
- Red Hat Enterprise Linux 2.1 Update 6
- SUSE LINUX Enterprise Server 9 Service Pack 1
- SUSE LINUX 9.3 (experimental support)
- SUSE LINUX 9.2

Support for Clustering with the iSCSI Protocol
Using the iSCSI protocol allows you to set up a clustering environment across hosts. For more information, see Clustering Using the iSCSI Protocol in the *VMware GSX Server Administration Guide*.

Dual-Core Processor Licensing Compatibility
The two processor version of GSX Server now supports systems with two dual-core processors.

Critical Bugs Fixed in GSX Server 3.2

Bugs fixed in this release include:

- Fix for panic that occurs on Red Hat Enterprise Linux 3.0 AS host.
- Fix for core dump when powering on virtual machine on Fedora Core 3 host. (Fedora is not a supported GSX Server host operating system.)
- Fix to allow USB devices to work in virtual machines on SUSE LINUX 9.1 host.
- Correction for monitor error that occurred when upgrading Windows Server 2003 guest operating system to Service Pack 1 RC1.
- Fix for inability to power on virtual machine with legacy virtual disk.
- Fix for DirectInput error that occurred when GSX Server automatically powered on a virtual machine when the Windows host starts. The message read:
  ```
  Failed to create DirectInput object-
  HRESULT(0x80004001:sev 1 fac 0 code 16385]:
  Unknown error 16385(0x4001)
  ```
- Fix for memory issue on hosts with 32GB RAM.
- Fix for VirtualCenter alarm showing maximum memory usage when virtual machine is idle or under low usage.
- Fix to prompt user to power off any running virtual machines before running the `vmware-config.pl` configuration program.
- Fix for security vulnerability described in CAN-2004-0700.
- Fix to improve scalability and performance on NUMA hardware.

Features in Version 3.1

Here are some highlights of key features to explore in VMware GSX Server 3.1:

Manage Virtual Disks with VMware Virtual Disk Manager
Create, manage and modify virtual disk files from the command line or within scripts with the VMware Virtual Disk Manager utility. For more information, see Using VMware Virtual Disk Manager on page 191.

New Operating System Support
Get the freedom to choose the operating systems and applications that work best for you. VMware GSX Server 3.1 provides experimental support for Microsoft Windows Server 2003 for 64-Bit Extended Systems (beta), Red Hat Enterprise Linux 3.0 and SuSE Linux Enterprise Server 8.0 on hosts with 64-bit processors.

Experimental guest operating system support is provided for Solaris 9 and 10 Operating System x86 Platform Edition.

VirtualCenter Ready
GSX Server 3.1 is enabled for management by VMware VirtualCenter. You can use VirtualCenter to manage and provision virtual machines on multiple GSX Server hosts, then migrate the virtual machines between other GSX Server and ESX Server systems under VirtualCenter management.

GSX Server 3.1 Secured with OpenSSL 0.9.7d
GSX Server 3.1 incorporates the updated version of OpenSSL. OpenSSL 0.9.7d corrects various security vulnerabilities.

Critical Bugs Fixed in GSX Server 3.1

Bugs fixed in this release include:

- Fix for Apache Web server crashes on Linux hosts after VMware Management Interface timed out.
- Fix for problem that prevented the launching of the VMware Virtual Machine Console from the VMware Management Interface on a Linux host.
- Fix for issue where the VMware Management Interface reported incorrect memory and processor usage statistics.
- Fix for VMware Management Interface issue where connections secured with SSL were redirected to insecure port 8222.
- Fix for Windows hosts where the Windows registry keys became read-only after removing GSX Server 2.x. This caused the installation of GSX Server 3 to fail.
- Fix for Linux hosts where users in particular groups could not create virtual machines.

- Fix for Linux hosts where switching to the `vmxnet` networking driver caused networking to fail in a virtual machine.
- Fix for Windows hosts where a virtual machines froze and the only workaround was to kill the virtual machine process.
- Fixes for SCSI reservation issues including deadlocks, and panics that occurred when powering on clustered virtual machines configured with LSI Logic SCSI adapters.
- Fix for issue that caused the VMware Registration Service service to hang when starting the host while virtual machines are configured to power on automatically with the host.
- Fix for an application error that occurred in Windows NT 4.0 Service Pack 6a (Japanese language version only) guest operating systems when using the search function of Windows Explorer. The error occurred only when VMware Tools is installed in the guest.
- Fix to allow the installation of VMware Tools in Windows NT 4.0 Support Pack 3 guest operating systems. For more information, see knowledge base article 1304 at *www.vmware.com/support/kb/enduser/std_adp.php?p_faqid=1304*.
- Fix for issue where a GSX Server 2.5.x virtual machine on a Windows host configured with a static IP address lost its static IP after it was powered on under GSX Server 3. For more information, see knowledge base article 1307 at *www.vmware.com/support/kb/enduser/std_adp.php?p_faqid=1307*.
- You can now restrict which users can create virtual machines and virtual disks. For more information, see knowledge base article 1042 at *www.vmware.com/support/kb/enduser/std_adp.php?p_faqid=1042*.
- Virtual machines configured for NAT networking can now use Point-to-Point Tunneling Protocol (PPTP).

Features in Version 3

Here are some highlights of key features to explore in VMware GSX Server 3:

GSX Server 3 Security Update: OpenSSL 0.9.7d Patches

Patches are available for GSX Server 3 that update the version of OpenSSL used in the product. The newer version of OpenSSL corrects security vulnerabilities and is posted on the GSX Server Security Updates Web page at *www.vmware.com/download/gsx_security.html*.

GSX Server 3 users are strongly urged to download and install these patches.

Enhanced Virtual Machine User Access Control
On Windows hosts, a virtual machine runs as a user. You can specify the particular user or let the virtual machine run as the user who powers it on. For more information, see Authenticating Users and Running Virtual Machines on a GSX Server for Windows Host in the *VMware GSX Server Administration Guide*.

Improved Security for Remote Connections
SSL is enabled by default for remote connections with the VMware Virtual Machine Console and the VMware Management Interface.

VirtualCenter Ready
GSX Server 3 is enabled for management by an upcoming release of VMware VirtualCenter. You will be able to use VirtualCenter to manage and provision virtual machines on multiple GSX Server hosts, then migrate the virtual machines between other GSX Server and ESX Server systems under VirtualCenter management.

New VMware Virtual Machine Console
Connect to and manage virtual machines with the VMware Virtual Machine Console, which combines the best abilities of the local and remote consoles in one application. Create and configure virtual machines locally and remotely. Connect to virtual machines from the local server or remote client at the same time, while other consoles are already connected. Run virtual machines in full screen mode locally and remotely.

The console interface is completely updated. You can run multiple virtual machines in the same window and switch from one to another using the new quick switch mode. The console menus have been streamlined. The console requires less network bandwidth over remote connections than the older console did.

For details, see Running Virtual Machines on page 95.

Take Snapshots of Your Virtual Machines
You can take a snapshot of your virtual machine's state, a point-in-time copy of the running system state, that is saved to disk. You can revert to that snapshot at any time — making it easier to do challenging tasks like upgrading guest operating systems. Take a snapshot, upgrade the operating system, and if something goes wrong, revert back to the snapshot. Or use the snapshot as a starting point for a sales demonstration.

See Taking Snapshots on page 153 for details.

Automatically Install GSX Server on Windows Hosts and VMware Tools in Windows Guests

We've improved the installers for Windows hosts and for VMware Tools in Windows guest operating systems. GSX Server for Windows hosts and VMware Tools for Windows guests use the Microsoft Windows Installer runtime engine, which allows you to automate the installation of GSX Server on a Windows host and VMware Tools in a Windows virtual machine. For information, see Automating the Installation of GSX Server in the *VMware GSX Server Administration Guide* and Automating the Installation of VMware Tools in a Windows Guest on page 69.

With the Microsoft Windows Installer runtime engine, you can pick and choose the features you want to install.

Automatically Start and Stop Virtual Machines When the Host Starts and Stops

You can configure virtual machines to automatically power on when the GSX Server host starts or automatically power off when the GSX Server host shuts down. For more information, see Powering Virtual Machines On and Off When the Host Starts and Shuts Down on page 122.

Connect to Older Virtual Machines and Older GSX Server Hosts

You can connect to older GSX Server hosts with consoles and run virtual machines from older versions of VMware products. For information, see Connecting to Older GSX Server and ESX Server Systems and Older Virtual Machines on page 113.

Improved VMware Management Interface

Managing virtual machines and the GSX Server host from a Web browser just got better. You can configure more host and virtual machine features, including virtual machine hardware, configuration options and SSL connections. For more information, see Using the VMware Management Interface in the *VMware GSX Server Administration Guide*.

Log GSX Server Events on Windows Hosts

GSX Server sends information about certain events that occur in the application on Windows hosts to the Windows Event Viewer. For details, see Logging GSX Server Events on Windows in the *VMware GSX Server Administration Guide*.

Improved Virtual Disk and Networking Performance

Experience 10 to 20% improvement in virtual disk and networking performance.

Monitor Virtual Machine Performance on Windows Hosts

Use GSX Server specific counters with the Windows Performance console (PerfMon) to monitor the performance of running virtual machines on Windows hosts. For

details, see Monitoring Virtual Machine Performance in the *VMware GSX Server Administration Guide*.

More Memory for Your Virtual Machines
Allocate up to 3600MB of memory to each virtual machine to run large server applications. For more information, see Allocating Memory to a Virtual Machine on page 340.

Easier Sharing of Virtual Machines with Latest VMware Products
Virtual machines created with GSX Server 3 are compatible with VMware Workstation 4 and ESX Server 2 for easier migrating and sharing of virtual machines.

New LSI Logic Virtual SCSI Adapter for Guest Operating Systems
The LSI Logic virtual SCSI adapter is included when you install Windows Server 2003, Red Hat Enterprise Linux 3.0 or NetWare guests.

PXE Boot
Boot virtual machines over your network and install guest operating systems from a PXE server. For more information, see Using PXE with Virtual Machines on page 132.

Easier Virtual Networking Configuration
On Windows hosts, use the Virtual Network Editor to configure virtual networking easily. For more information, see Networking on page 227.

Network Adapter Teaming Support
On Windows hosts, virtual machines can bridge to teamed or bonded host network adapters. For more information, see Configuring Bridged Networking When Using Teamed Network Interface Cards on Your Host on page 259.

Generic SCSI Tape Backup Support
Back up virtual machines using popular backup software and SCSI tape devices. For more information, see Backing Up Virtual Machines and the GSX Server Host in the *VMware GSX Server Administration Guide*.

Using DVD-ROM and CD-ROM Drives on Remote Clients
If you're connected to a virtual machine remotely from a client, you can use the local DVD-ROM or CD-ROM drive to install software or copy data without needing to use the drive on the GSX Server host. For more information, see Using the DVD-ROM or CD-ROM Drive on a Client on page 173.

Improved Virtual DVD-ROM and CD-ROM Drive Support
Read multisession DVD-ROM and CD-ROM media. Burn CD-ROMs in your guest operating systems.

Debugging Support in Virtual Machines
GSX Server supports user- and kernel-level debuggers in virtual machines.

New Operating System Support
VMware GSX Server 3 provides support for Red Hat Enterprise Linux 3.0, SuSE Linux Enterprise Server 8.0 patch 3, and Turbolinux Server 8.0 and Workstation 8.0 hosts. New supported guest operating systems include Red Hat Enterprise Linux 3.0; SuSE Linux Enterprise Server 7 patch 2; NetWare 6.5 Server; FreeBSD 4.6.2, 4.8, 5.0 and 5.1 (prerelease version); and Turbolinux Server 7.0, 8.0 and Workstation 8.0. Experimental support for Microsoft Windows code-named Longhorn is provided.

New Linux Kernel Support
Run your Linux guest operating systems with the new 2.6 kernel.

New Support Scripts
When you file support requests, please use the new support scripts to collect data that help us diagnose your problems. For details, see Reporting Problems on page 29.

Automatically Check for Product Updates
VMware GSX Server now checks automatically to see if updates for the product are available. You can specify what interval to use for the automatic check or switch to manual checks only. For more information, see Updating GSX Server Software Automatically in the *VMware GSX Server Administration Guide*.

Virtual Machine Specifications

Each virtual machine created with GSX Server provides a platform that includes the following devices that your guest operating system can see.

Virtual Processor

- Same processor as that on host computer (but no extended 64-bit support available)
- Single processor per virtual machine on symmetric multiprocessor (SMP) systems

Virtual Chip Set

- Intel 440BX-based motherboard with NS338 SIO chip and 82093AA IOAPIC

Virtual BIOS

- PhoenixBIOS™ 4.0 Release 6 with VESA BIOS
- DMI/SMBIOS-compliant for system management agent support

Virtual Memory

- Up to 3600MB of memory per virtual machine, depending upon the host system's configuration, the types of applications running on the host and the amount of memory on the host

Virtual Graphics

- VGA and SVGA support

Virtual IDE Drives

- Up to four devices — disks, CD-ROM or DVD-ROM (DVD drives can be used to read data DVD-ROM discs; DVD video is not supported)
- Hard disks can be virtual disks or physical disks
- IDE virtual disks up to 128GB
- CD-ROM can be a physical device or an ISO image file

Virtual SCSI Devices

- Up to 60 devices on up to four virtual SCSI controllers
- SCSI virtual disks up to 256GB
- Hard disks can be virtual disks or physical disks
- Generic SCSI support allows scanners, CD-ROM, DVD-ROM, tape drives and other SCSI devices to be used without requiring drivers in the host operating system

- Mylex® (BusLogic) BT-958 compatible host bus adapter
- LSI Logic Ultra160 LSI53C10xx SCSI controller

Virtual PCI Slots

- Six virtual PCI slots, to be divided among the virtual SCSI controllers, virtual Ethernet cards, virtual display adapter and virtual sound adapter

Virtual Floppy Drives

- Up to two 1.44MB floppy devices
- Physical drives or floppy image files

Virtual Serial (COM) Ports

- Up to four serial (COM) ports
- Output to serial ports, Windows files, Linux files or named pipes

Virtual Parallel (LPT) Ports

- Up to three bidirectional parallel (LPT) ports
- Output to parallel ports or host operating system files

Virtual USB ports

- Two-port USB 1.1 UHCI controller
- Supported devices include USB printers, scanners, PDAs, hard disk drives, memory card readers and still digital cameras

Virtual Keyboard

- 104-key Windows 95/98 enhanced

Virtual Mouse and Drawing Tablets

- PS/2 mouse
- Serial tablet support

Virtual Ethernet Card

- Up to four virtual Ethernet cards
- AMD PCnet-PCI II compatible
- Wireless networking support with bridged and NAT networking
- PXE ROM version 2.0

Virtual Networking

- Nine virtual Ethernet switches (three configured by default for bridged, host-only and NAT networking)

- Virtual networking supports most Ethernet-based protocols, including TCP/IP, NetBEUI, Microsoft Networking, Samba, Novell® NetWare® and Network File System
- Built-in NAT supports client software using TCP/IP, FTP, DNS, HTTP and Telnet

Virtual Sound Adapter

- Sound output and input
- Creative Labs Sound Blaster® AudioPCI emulation (MIDI input, game controllers and joysticks are not supported)

Supported Guest Operating Systems

The operating systems listed here have been tested in VMware GSX Server virtual machines and are officially supported. For notes on installing guest operating systems, see the *VMware Guest Operating System Installation Guide*, available from the Help menu or from the VMware Web site at *www.vmware.com/support/guestnotes/doc/index.html*.

Operating systems that are not listed are not supported for use in a VMware GSX Server virtual machine. For the most recent list of supported guest operating systems, visit the VMware Web site at *www.vmware.com/support/gsx3/doc/intro_sysreqs_guest_gsx.html*.

Note: Operating systems that are not listed are not supported for use in a VMware GSX Server virtual machine. Guest operating systems using 64-bit extensions to the IA-32 instruction set are not supported.

Microsoft Windows

- Microsoft Windows code-named Longhorn (experimental support)
- Windows Server 2003 Web Edition, Windows Server 2003 Standard Edition, Windows Server 2003 Enterprise Edition and Windows Small Business Server 2003, including Service Pack 1
- Windows XP Professional and Windows XP Home Edition, including Service Pack 1 and Service Pack 2
- Windows 2000 Professional, Windows 2000 Server and Windows 2000 Advanced Server, including Service Pack 1, Service Pack 2, Service Pack 3 and Service Pack 4, and Windows 2000 Professional Service Pack 4 checked build
- Windows NT® 4.0 Server Service Pack 6a, Windows NT Workstation 4.0, including Service Pack 6a and Windows NT 4.0 Terminal Server Edition Service Pack 6a
- Windows Me
- Windows 98, including latest Customer Service Packs, and Windows 98 SE
- Windows 95, including Service Pack 1 and all OSR releases
- Windows for Workgroups 3.11
- Windows 3.1

Microsoft MS-DOS

- MS-DOS 6.22

Linux

- Mandrake Linux 8.0, 8.1, 8.2, 9.0, 9.1, 9.2, 10.0 and 10.1
- Red Hat Linux 6.2, 7.0, 7.1, 7.2, 7.3, 8.0 and 9.0
- Red Hat Enterprise Linux (AS, ES and WS) 2.1, including Update 6; Red Hat Enterprise Linux (AS, ES and WS) 3.0, including Update 4; Red Hat Enterprise Linux (AS, ES and WS) 4.0
- SuSE Linux 7.3, 8.0, 8.1, 8.2, 9.0, 9.1 and 9.2; experimental support for SuSE Linux Enterprise Server 7, including Service Pack 2; 8, including Service Pack 3; and 9, including Service Pack 1
- Turbolinux Server 7.0, 8.0, and Workstation 8.0

Novell NetWare

- NetWare 4.2 Support Pack 9, 5.1 Support Pack 6, 6.0 Support Pack 3 and 6.5 Support Pack 1

FreeBSD

- FreeBSD 4.0–4.6.2, 4.8, 4.9, 5.0, 5.1 and 5.2

Solaris

- Solaris 9 and 10 Operating System x86 Platform Edition (experimental support)

Technical Support Resources

The following sections describe various technical support resources available to you.

- Self-Service Support
- Online and Telephone Support
- Support Offerings
- Reporting Problems
- Log Files

Self-Service Support

Use the VMware Technology Network for self help tools and technical information:

- Product Information — *www.vmware.com/support/resources*
- Technology Information — *www.vmware.com/vcommunity/technology*
- Documentation — *www.vmware.com/support/pubs*
- Knowledge Base — *www.vmware.com/support/kb*
- Discussion Forums — *www.vmware.com/community*
- User Groups — *www.vmware.com/vcommunity/usergroups.html*

For more information about the VMware Technology Network, go to *www.vmtn.net*.

Online and Telephone Support

Use online support to submit technical support requests, view your product and contract information, and register your products. Go to *www.vmware.com/support*.

Use phone support for the fastest response on priority 1 issues for customers with appropriate support contracts. Go to *www.vmware.com/support/phone_support.html*.

Support Offerings

Find out how VMware's support offerings can help you meet your business needs. Go to *www.vmware.com/support/services*.

Reporting Problems

If you have problems while running GSX Server, please report them to the VMware support team.

You must register your serial number; then you can report your problems by submitting a support request at *www.vmware.com/requestsupport*.

The steps below describe the information we need from you to diagnose problems. This information largely comes from various log files. Which log file we need depends upon the problem you encounter. The log files are listed after the steps.

You can simplify the process of collecting the needed information by running the support script to collect the appropriate log files and system information. Follow the steps below that apply to your host computer.

Note: The support script runs only on the GSX Server host. If you encounter problems on a remote client, you must supply the log files manually. The two log files you should supply, depending upon the problem you encounter on the client, include the VMware Virtual Machine Console log file and the installation log file. See below for more information about these logs.

Windows Host

1. Open a command prompt.
2. Change to the GSX Server program directory.
   ```
   C:
   cd \Program Files\VMware\VMware GSX Server
   ```
   If you did not install the program in the default directory, use the appropriate drive letter and substitute the appropriate path in the `cd` command above.
3. Run the support script.
   ```
   cscript vm-support.vbs
   ```
4. After the script runs, it displays the name of the directory where it has stored its output. Use a file compression utility such as WinZip or PKZIP to zip that directory, then include the zip file with your support request.

Linux Host

1. Open a terminal.
2. Run the support script as the user who is running the virtual machine or as root.
   ```
   vm-support
   ```
   If you do not run the script as root, the script displays messages indicating that it cannot collect some information. This is normal. If the VMware support team needs that information, a support representative may ask you to run the script again as root.
3. The script creates a compressed `.tgz` file in the current directory. Include that output file with your support request.

Log Files

The following log files are generated by GSX Server and are collected by the support script as needed. Since there is no support script on a remote client, you need to submit a support request at *www.vmware.com/requestsupport* for any issues you encounter on a client and include the console's log file or its installation log file.

Virtual Machine Log File

If a virtual machine exits abnormally or crashes, please run the support script or save the log file before you launch that virtual machine again. The key log file to save is the VMware log file for the affected virtual machine.

On a Windows host, the `vmware.log` file is in the same directory as the configuration file (`.vmx`) of the virtual machine that had problems. The path to the log file of the active virtual machine appears in the About dialog box. In a console, choose **Help** > **About VMware GSX Server**, and look under **Additional information**.

On a Linux host, the `<vmname>.log` file is in the same directory as the configuration file (`.vmx`) of the virtual machine that had problems.

Also save any core files (`core` or `vmware-core`).

Virtual Machine Event Log File

The virtual machine's event log, some of which can be viewed in the VMware Management Interface, is stored as a file on the host. This file can also be useful in the event a virtual machine crashes.

Each virtual machine on the host includes an event log file called `event-<path_to_configuration_file>.vmx.log`.

On a Windows host, the log is stored in `C:\Program Files\VMware\VMware GSX Server\vmserverdRoot\eventlog`.

On a Linux host, the log is stored in `/var/log/vmware`.

VMware Virtual Machine Console Log File

The VMware Virtual Machine Console keeps a log. If you encounter problems with the VMware Virtual Machine Console on a remote client, please submit a support request and this log file.

On a Windows host, the log is called `vmware-<username>-<PID>.log` and is stored in the user's TEMP directory; by default, this directory is `C:\Documents and Settings\<username>\Local Settings\Temp`. The path to this file appears in the About dialog box. In a console, choose **Help** > **About VMware GSX Server**, and look under **Additional information**.

On a Linux host, the log is called `ui-<PID>.log` and is stored in the user's TEMP directory; by default, this directory is `/tmp/vmware-<username>`. The path to this file appears in the terminal when you start the console.

VMware Management Interface Log File

The VMware Management Interface keeps a log.

On a Windows host, the log is called `mui.log` and is stored by default in `C:\Program Files\VMware\VMware Management Interface`.

On a Linux host, the log is called `error_log` and is stored by default in `/var/log/vmware-mui`.

VMware Authorization Service Log File

You can enable logging for the VMware Authorization Service (known as `vmware-authd` on Linux hosts) manually.

1. In a text editor, open the following file:
   - On a Windows host, edit `config.ini`, located in `C:\Documents and Settings\All Users\Application Data\VMware\VMware GSX Server`.
   - On a Linux host, edit `/etc/vmware/config`.
2. Add the following lines to the file:
   ```
   vmauthd.logEnabled = TRUE
   log.vmauthdFileName = "vmauthd.log"
   ```
   This creates a file called `vmauthd.log`. On a Windows host, this file appears by default in `C:\Windows\system32` or `C:\WINNT\system32`; on a Linux host, this file appears by default in `/var/log/vmware`.
3. Save and close the configuration file. The log is enabled on a Linux host.
4. On a Windows host, restart the VMware Authorization Service. Choose **Start** > **Administrative Tools** > **Services**. Right-click **VMware Authorization Service** and choose **Restart**. This enables logging.

VMware Registration Service Log File

The VMware Registration Service keeps a log.

On a Windows host, the log is called `vmware-serverd.log` and is stored in `C:\Windows\Temp`.

On a Linux host, the log is called `vmware-serverd.log` and is stored in `/var/log/vmware`.

VMware GSX Server and VMware Virtual Machine Console Installation Log Files

GSX Server keeps an installation log file on the server host.

On a remote client, the VMware Virtual Machine Console keeps an installation log file. If you encounter problems installing the VMware Virtual Machine Console, please submit a support request and this log file.

On a Windows host, the file is `VMInst.log`. It is saved in your TEMP directory; the default location is `C:\Documents and Settings\<username>\Local Settings\Temp`. The `Local Settings` folder is hidden by default. To see its contents, open **My Computer**, choose **Tools** > **Folder Options**, click the **View** tab and select **Show Hidden Files and Folders**.

On a Linux host, the log is called `locations` and is stored in `/etc/vmware`.

CHAPTER 2

Creating a New Virtual Machine

The following sections describe how to create a new virtual machine:

- Setting Up a New Virtual Machine on page 36
- Installing a Guest Operating System on page 56

Setting Up a New Virtual Machine

The New Virtual Machine Wizard guides you through the key steps for setting up a new virtual machine, helping you set various options and parameters. You can then use the virtual machine settings editor (**VM** > **Settings**) if you need to make any changes to your virtual machine's setup.

- To create a new virtual machine from a console, see Creating a New Virtual Machine with the New Virtual Machine Wizard on page 38.
- To create a new virtual machine from the VMware Management Interface, see Creating a New Virtual Machine from the VMware Management Interface on page 50.
- To create a new virtual machine on the GSX Server host from a VirtualCenter client, see Creating Virtual Machines on a GSX Server Host from a VirtualCenter Client in the *VMware GSX Server Administration Guide*.

What's in a Virtual Machine?

The virtual machine typically is stored on the host computer in a set of files, all of which are in a directory set aside for that particular virtual machine. In these examples, `<vmname>` is the name of your virtual machine. The key files are:

- `<vmname>.vmx` — the configuration file, which stores settings chosen in the New Virtual Machine Wizard or virtual machine settings editor. If you created the virtual machine under an earlier version of VMware GSX Server on a Linux host, this file may have a `.cfg` extension.
- `nvram` — the file that stores the state of the virtual machine's BIOS.
- `<vmname>.vmdk` — the virtual disk file, which stores the contents of the virtual machine's hard disk drive.

  A virtual disk is made up of one or more `.vmdk` files. If you have specified that the virtual disk should be split into 2GB files, the number of `.vmdk` files depends on the size of the virtual disk.

  By default, all virtual disk space is preallocated when you create the virtual disk. Make sure you have enough disk space on the host before you create a preallocated disk.

  If you decide to not allocate all disk space when you create the virtual disk, the `.vmdk` files grow in size as data is added to the virtual disk. Almost all of a `.vmdk` file's content is the virtual machine's data, with a small portion allotted to virtual machine overhead.

If the virtual machine is connected directly to a physical disk, rather than to a virtual disk, the `.vmdk` file stores information about the partitions the virtual machine is allowed to access.

Note: Earlier VMware products used the extension `.dsk` for virtual disk files.

- `<vmname>.log` or `vmware.log` — the file that keeps a log of key virtual machine activity. This file can be useful in troubleshooting if you encounter problems. This file is stored in the directory that holds the configuration file (`.vmx`) of the virtual machine.
- `<vmname>.vmdk.REDO_xxxxxx` — a redo-log file, created automatically when a virtual machine has a snapshot or is in independent-nonpersistent mode. This file stores changes made to a virtual disk while the virtual machine is running. There may be more than one such file. The `xxxxxx` indicates a unique suffix added automatically by GSX Server to avoid duplicate filenames.
- `<vmname>.vmss` — the suspended state file, which stores the state of a suspended virtual machine.

  Note: Some earlier VMware products used the extension `.std` for suspended state files.
- `<vmname>.vmsn` — the snapshot state file, which stores the running state of a virtual machine at the time you take a snapshot of it.
- `<vmname>.vmx.sav` — the configuration snapshot file, which stores the configuration of a virtual machine at the time you take a snapshot of it. If you created the virtual machine under an earlier version of GSX Server on a Linux host, this file may have a `.cfg` extension.

There may be other files as well, some of which are present only while a virtual machine is running.

Permissions and Running Virtual Machines

When you create a virtual machine, by default the virtual machine is private, which means you are the only user who can access it. If you choose the custom path when creating the virtual machine, you can specify that all users can access the virtual machine.

When a virtual machine is private, it appears in the inventory of the console of the user who created it. The virtual machine does not appear in the inventory of consoles for other users connected to the host. The virtual machine appears in the VMware Management Interface only when you are logged on as the user who created the virtual machine.

When the virtual machine is running, the actions you can take with it depend upon your permissions. For more information about permissions, see Understanding Permissions and Virtual Machines in the *VMware GSX Server Administration Guide*.

Creating a New Virtual Machine with the New Virtual Machine Wizard

When you create a new virtual machine, you end up with a set of files that represent a new computer, complete with a blank, unformatted hard disk — the virtual disk — onto which you install the guest operating system. The virtual disk by default has all its disk space preallocated at the time it is created.

The virtual machines you create are located on the host to which you are currently logged on, even if the console you are using is running on a remote client.

Note: Before you create the virtual machine, you should check the installation notes for the guest operating system you intend to install in it. You can find this information in the *VMware Guest Operating System Installation Guide* available from the Help menu or from the VMware Web site at *www.vmware.com/support/guestnotes/doc/index.html*.

Complete the following steps to create a new virtual machine.

1. Launch the VMware Virtual Machine Console.

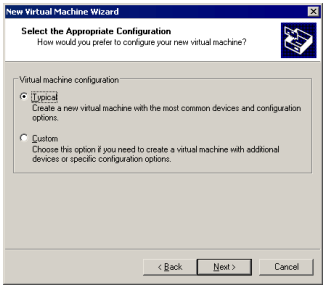
   Windows hosts: See Connecting to a Virtual Machine from a Windows Host or Client on page 107.

   Linux hosts: See Connecting to a Virtual Machine from a Linux Host or Client on page 110.

2. Start the New Virtual Machine Wizard. Choose **File** > **New Virtual Machine** or click the New Virtual Machine icon on the console **Home** tab.

   The New Virtual Machine Wizard presents you with a series of screens that you navigate using the **Next** and **Back** buttons at the bottom of each screen. At each screen, follow the instructions, then click **Next** to proceed to the next screen.

3. Select the method you want to use for configuring your virtual machine.


New Virtual Machine Wizard
Select the Appropriate Configuration
How would you prefer to configure your new virtual machine?
Virtual machine configuration
Typical
Create a new virtual machine with the most common devices and configuration options.
Custom
Choose this option if you need to create a virtual machine with additional devices or specific configuration options.
< Back
Next >
Cancel

If you select **Typical**, you can specify or accept defaults only for

- The guest operating system.
- The virtual machine name and the location of the virtual machine's files.
- The network connection type.
- The size of the virtual disk.
- Allocating all the disk space for the virtual disk at the time you create it.
- Splitting the virtual disk into 2GB files.

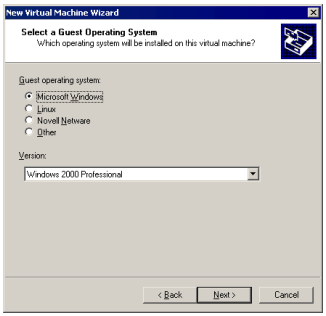
Select **Custom** if you want to

- Allocate an amount of memory different from the default.
- Choose between the LSI Logic and BusLogic types of SCSI adapters. (An ATAPI IDE adapter is always installed.)
- Let other users access this virtual machine.
- Have the virtual machine automatically power on or off when the GSX Server Windows host starts up or shuts down.
- Specify the user account the virtual machine uses when running.
- Use an existing virtual disk.
- Use a physical disk rather than a virtual disk (for advanced users).
- Use an IDE virtual disk for a guest operating system that would otherwise have a SCSI virtual disk created by default and vice versa.
- Create a virtual disk as a single disk file. If the virtual disk is larger than 2GB, the host file system must support files larger than 2GB.
- Store your virtual disk files in a particular location.
- Specify a particular virtual device node for the virtual disk.

- Use independent disk mode (if you don't plan to use snapshots with this virtual machine; see Independent Disks on page 164).

Note: If you follow the custom path, you still specify the options under the typical path.

4. Select a guest operating system.



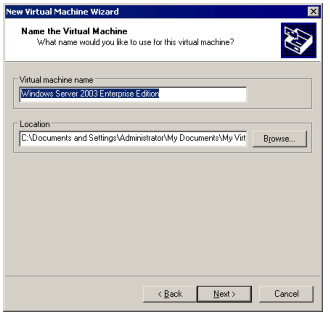
This screen asks which operating system you plan to install in the virtual machine. The New Virtual Machine Wizard uses this information to select appropriate default values, such as the amount of memory needed. The wizard also uses this information when naming associated virtual machine files.

Under **Guest operating system**, select the operating system family (Microsoft Windows, Linux, Novell NetWare or Other — for MS-DOS, FreeBSD or other guests not listed), then select the specific operating system from the **Version** list.

If the operating system you are using is not listed, select **Other**, then select **Other** again in the **Version** list.

The remaining steps assume you plan to install a Windows Server 2003 Enterprise guest operating system. You can find detailed installation notes for this and other guest operating systems in the *VMware Guest Operating System Installation Guide*, available from the Help menu or from the VMware Web site at *www.vmware.com/support/guestnotes/doc/index.html*.

5. Select a name and directory for the virtual machine.



The name specified here is used in the VMware Virtual Machine Console and the VMware Management Interface. It is also used as the name of the directory where the files associated with this virtual machine are stored.

Each virtual machine must have its own directory. All associated files, such as the configuration file and the disk file, are placed in this directory.

Windows hosts: The virtual machine directory and its files are stored in the default location `<installdrive>:\Virtual Machines`.

Linux hosts: The virtual machine directory and its files are stored in the default location `/var/lib/vmware/Virtual Machines`.

If some users without access to this host need to access this virtual machine, you may consider placing the virtual machine files in a location that is accessible to them. For more information, see Sharing Virtual Machines with Other Users in the *VMware GSX Server Administration Guide*.

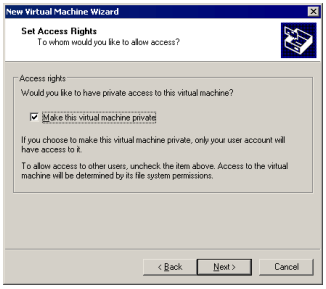
Note: You can change the default location from the console; choose **Host** > **Settings** > **General**. Click **Browse** to select a new path. Make sure that you locate the virtual machine in a unique directory.

Virtual machine performance may be slower if your virtual hard disk is on a network drive. For best performance, be sure the virtual machine's directory is on a local drive. However, if other users need to access this virtual machine, you should consider placing the virtual machine files in a location that is accessible to them. For more information, see Sharing Virtual Machines with Other Users in the *VMware GSX Server Administration Guide*.

If you selected **Typical** as your configuration path, go to step 9.

If you selected **Custom** as your configuration path, continue with the steps for customizing your virtual machine configuration.

6. Specify whether this virtual machine should be private.


New Virtual Machine Wizard
Set Access Rights
To whom would you like to allow access?
Access rights
Would you like to have private access to this virtual machine?
Make this virtual machine private
If you choose to make this virtual machine private, only your user account will have access to it.
To allow access to other users, uncheck the item above. Access to the virtual machine will be determined by its file system permissions.
< Back
Next >
Cancel

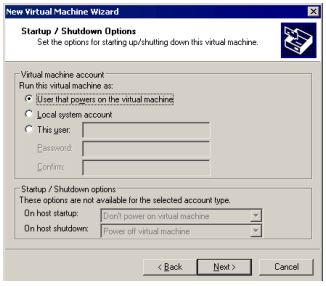
By default, a virtual machine is private, so only you have access to it. This is useful, for example, if you are in charge of provisioning virtual machines on one host and will propagate the virtual machines to other hosts.

If you are creating a virtual machine using the **Typical** path, then only you can access the virtual machine. The virtual machine is private.

You can change access to this virtual machine in the virtual machine settings editor (choose **VM** > **Settings** > **Options** > **Permissions**). For more information about private virtual machines, see Only You Can See Virtual Machines You Create in the *VMware GSX Server Administration Guide*.

If you are a host administrator, you can specify virtual machine permissions in the host configuration. For more information about permissions and virtual machines, see Securing Virtual Machines and the Host in the *VMware GSX Server Administration Guide*.

7. Choose the user account for running the virtual machine (for virtual machines on Windows hosts only) and the host startup and shutdown options.


New Virtual Machine Wizard
Startup / Shutdown Options
Set the options for starting up/shutting down this virtual machine.
Virtual machine account
Run this virtual machine as:
User that powers on the virtual machine
Local system account
This user:
Password:
Confirm:
Startup / Shutdown options
These options are not available for the selected account type.
On host startup:
Don't power on virtual machine
On host shutdown:
Power off virtual machine
< Back
Next >
Cancel

Windows hosts: Under **Virtual machine account**, choose which user account the virtual machine uses when it runs. This account is used for actions like

network access from within the virtual machine and access to virtual machine resources that are on the network.

- **User that powers on the virtual machine** — the virtual machine runs as the account of the user who powered on the virtual machine until the virtual machine is powered off. Other users can connect to the virtual machine but it still runs as the user who powered on the virtual machine.

  The level of access other users have to this virtual machine is based on the level of access of the user that powers it on. For information about user access to virtual machines, see Understanding Permissions and Virtual Machines in the *VMware GSX Server Administration Guide*.

- **Local system account** — the virtual machine runs as the local system account (administrator). You can enable this option only if you are logged on to the host operating system as an administrator.

  Note: This user can run virtual machines that are in local storage only.

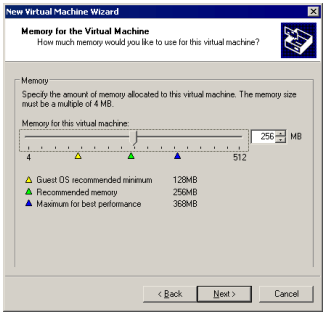
- **This user** — the virtual machine runs as the user account specified here. The password is not validated until you power on the virtual machine. You can specify a local user account, a local system administrator account or a fully-qualified domain user account for this user.

All hosts: Under **Startup/Shutdown Options**, choose whether you want this virtual machine to power on automatically when the GSX Server host starts up and whether you want the virtual machine to power off when the host shuts down. You can configure a virtual machine to start up or shut down automatically when the host starts or shuts down only when the host is configured accordingly. If the host settings are disabled, you must enable them before you can specify these options for a virtual machine. For more information, see Configuring Startup and Shutdown Options for Virtual Machines in the *VMware GSX Server Administration Guide*.

To enable the startup and shutdown options, you must configure the virtual machine to run as an administrator user.

When the virtual machine is powered off, you can change all these options.

8. Allocate an amount of memory to the virtual machine.


New Virtual Machine Wizard
Memory for the Virtual Machine
How much memory would you like to use for this virtual machine?
Memory
Specify the amount of memory allocated to this virtual machine. The memory size must be a multiple of 4 MB.
Memory for this virtual machine:
256
MB
4
512
Guest OS recommended minimum 128MB
Recommended memory 256MB
Maximum for best performance 368MB
< Back
Next >
Cancel

The New Virtual Machine Wizard provides a default value based on your guest operating system selection, along with the recommended range and the total amount of memory all running virtual machines can use.

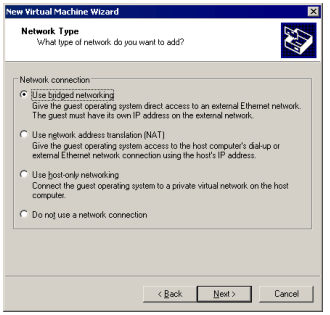
The wizard also indicates the minimum amount of memory recommended by the manufacturer and the GSX Server recommended maximum value for best performance of your virtual machine on this server host.

To change the amount of memory to be allocated to the virtual machine, move the slider to the appropriate location, use the spin controller next to the field or type a new value in the field.

Caution: You cannot allocate more than 2000MB of memory to a virtual machine if it is stored on a file system that cannot support files larger than 2GB, such as FAT16. You will not be able to power on such a virtual machine. Further, you cannot allocate more than 2000MB of memory to a virtual machine if it is stored on a FAT32 file system, even though it does support files up to 4GB in size.

For more information about memory, see Understanding Memory Usage in the *VMware GSX Server Administration Guide* and Allocating Memory to a Virtual Machine on page 340.

9. Configure the networking capabilities of the virtual machine.


New Virtual Machine Wizard
Network Type
What type of network do you want to add?
Network connection
Use bridged networking
Give the guest operating system direct access to an external Ethernet network. The guest must have its own IP address on the external network.
Use network address translation (NAT)
Give the guest operating system access to the host computer's dial-up or external Ethernet network connection using the host's IP address.
Use host-only networking
Connect the guest operating system to a private virtual network on the host computer.
Do not use a network connection
< Back
Next >
Cancel

If your host computer is on a network and you have a separate IP address for your virtual machine (or can get one automatically from a DHCP server), select **Use bridged networking**.

If you do not have a separate IP address for your virtual machine but you want to be able to connect to the Internet, select **Use network address translation (NAT)**. NAT is useful if you have a wireless network adapter on a Linux host (as bridged networking on wireless network adapters is supported only on Windows hosts). It also allows for the sharing of files between the virtual machine and the host operating system.

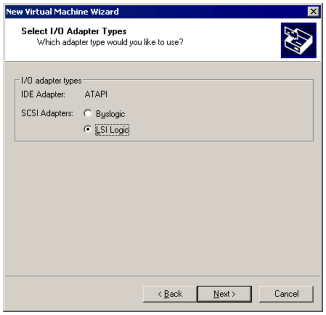
To enable your virtual machine to use a virtual network limited to the host and the virtual machines on the host using only the host-only network adapter, select **Use host-only networking**.

For more details about VMware GSX Server networking options, see Networking on page 227.

If you selected **Typical** as your configuration path, go to step 13.

If you selected **Custom** as your configuration path, continue with the steps for customizing your virtual machine configuration.

10. Choose the type of SCSI adapter you want to use with the virtual machine.



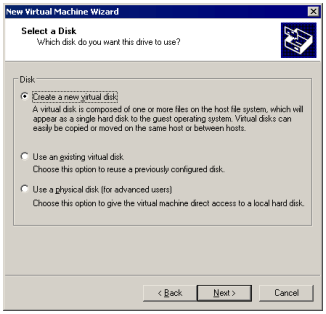
An IDE and a SCSI adapter are installed in the virtual machine. The IDE adapter is always ATAPI. You can choose between a BusLogic or LSI Logic SCSI adapter. The default for your guest operating system is already selected. Most guests except for newer operating systems like Windows Server 2003, Red Hat Enterprise Linux 3 and NetWare 6.5 default to the BusLogic adapter.

The LSI Logic adapter has improved performance and works better with generic SCSI devices. The LSI Logic adapter is included with Windows Server 2003.

The choice of SCSI adapter does not affect your decision to make your virtual disk an IDE or SCSI disk. However, most guest operating systems do not include a driver for the LSI Logic adapter; you must download the driver from the LSI Logic Web site. See the *VMware Guest Operating System Installation Guide* for details about the driver and the guest operating system you plan to install in this virtual machine.

You cannot change the SCSI adapter type after you create the virtual machine.

11. Select the disk you want to use with the virtual machine.


New Virtual Machine Wizard
Select a Disk
Which disk do you want this drive to use?
Disk
Create a new virtual disk
A virtual disk is composed of one or more files on the host file system, which will appear as a single hard disk to the guest operating system. Virtual disks can easily be copied or moved on the same host or between hosts.
Use an existing virtual disk
Choose this option to reuse a previously configured disk.
Use a physical disk (for advanced users)
Choose this option to give the virtual machine direct access to a local hard disk.
< Back
Next >
Cancel

To use a new, unformatted virtual disk, select **Create a new virtual disk**.

Virtual disks are the best choice for most virtual machines. They are quick and easy to set up and can be moved to new locations on the same host computer or to different host computers.

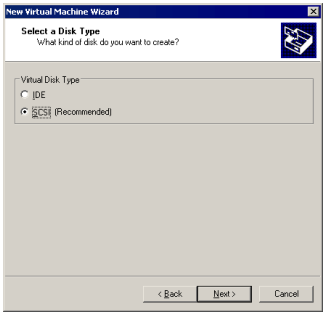
To use an existing virtual disk with this virtual machine, select **Use an existing virtual disk**. Browse to select the disk.

To install the guest operating system on a physical (also called raw) IDE disk, select **Use a physical disk**. To use a physical SCSI disk, add it to the virtual machine later with the virtual machine settings editor (**VM** > **Settings**). Booting from a physical SCSI disk is not supported.

To install your guest operating system directly on an existing IDE disk partition, read the reference note Installing an Operating System onto a Physical Partition from a Virtual Machine on page 220.

Caution: VMware recommends that only advanced users should use physical disks with virtual machines.

12. Select whether you want the virtual disk to be an IDE disk or a SCSI disk.

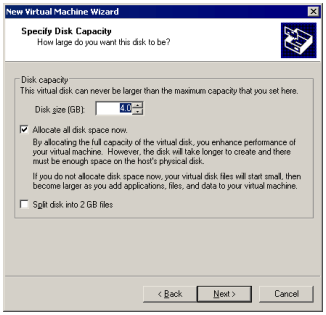


The wizard recommends the best choice based on the guest operating system you selected.

Make the Virtual Disk Big Enough

- The virtual disk should be large enough to hold the guest operating system and all of the software that you intend to install, with room for data and growth.
- You cannot change the virtual disk's maximum capacity later.
- You can install additional virtual disks using the virtual machine settings editor.
- For example, you need about 1GB of actual free space on the file system containing the virtual disk to install Windows Server 2003 and applications such as Microsoft Office inside the virtual machine. You can set up a single virtual disk to hold these files. Or you can split them up — installing the operating system on the first virtual disk and using a second virtual disk for applications or data files.

13. Specify the capacity of the virtual disk.


New Virtual Machine Wizard
Specify Disk Capacity
How large do you want this disk to be?
Disk capacity
This virtual disk can never be larger than the maximum capacity that you set here.
Disk size (GB):
4.0
Allocate all disk space now.
By allocating the full capacity of the virtual disk, you enhance performance of your virtual machine. However, the disk will take longer to create and there must be enough space on the host's physical disk.
If you do not allocate disk space now, your virtual disk files will start small, then become larger as you add applications, files, and data to your virtual machine.
Split disk into 2 GB files
< Back
Next >
Cancel

Enter the size of the virtual disk that you wish to create.

Your virtual disk can be as small as 0.1GB (100MB). A SCSI virtual disk can be as large as 256GB; an IDE virtual disk can be as large as 128GB. The default is 4GB.

By default, the full size of the virtual disk is allocated when you create the disk. Allocating all the space at the time you create the virtual disk gives somewhat better performance and ensures you do not run out of disk space on the host, but it requires as much disk space as the size you specify for the virtual disk. You cannot shrink a preallocated disk.

If this setting is larger than the space available on the host machine's hard disk, a warning message appears, and specifies how much space you have on the host. If the disk will exceed the available space on the host, you must make the virtual disk smaller or clear the **Allocate all disk space now** check box.

A preallocated virtual disk is needed for clustering virtual machines. For more information about clustering, see High-Availability Configurations with VMware GSX Server in the *VMware GSX Server Administration Guide*.

If you do not preallocate the disk, the virtual disk's files start small and grow as needed.

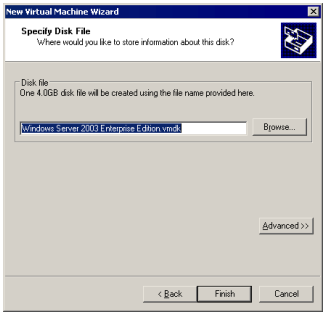
You may also specify whether you want the virtual disk created as one large file or split into a set of 2GB files. You should split the virtual disk if it is stored on a FAT32 file system or a file system that cannot support files larger than 2GB, such as FAT16. To do this, check **Split into 2GB files**.

If you later decide you want to enlarge a virtual disk, change the virtual disk from preallocated to growable (or vice versa), or split a virtual disk that was created in a single file (or vice versa), you can use the VMware Virtual Disk Manager to

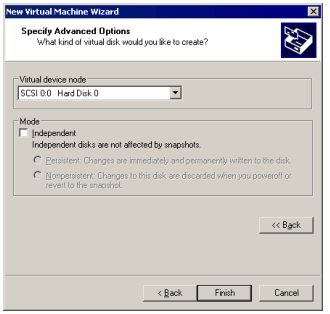
expand or convert the virtual disk. For more information, see Using VMware Virtual Disk Manager on page 191.

If you selected **Typical** as your configuration path and you have set the disk options you want to use, click **Finish**. GSX Server creates the virtual machine. If you selected **Custom** as your configuration path, continue to the next step.

14. Specify the name and location of the virtual disk's files.


New Virtual Machine Wizard
Specify Disk File
Where would you like to store information about this disk?
Disk file
One 4.0GB disk file will be created using the file name provided here.
Windows Server 2003 Enterprise Edition.vmdk
Browse...
Advanced >>
< Back
Finish
Cancel

If you want to specify which virtual device node should be used by your virtual disk or if you want to use independent disk mode, click **Advanced**.


New Virtual Machine Wizard
Specify Advanced Options
What kind of virtual disk would you like to create?
Virtual device node
SCSI 0:0 Hard Disk 0
Mode
Independent
Independent disks are not affected by snapshots.
Persistent: Changes are immediately and permanently written to the disk.
Nonpersistent: Changes to this disk are discarded when you poweroff or revert to the snapshot.
<< Back
< Back
Finish
Cancel

Specifying a disk mode is useful in certain special-purpose configurations in which you want to exclude disks from the snapshot. For more information on the snapshot feature, see Taking Snapshots on page 153.

Normal disks are included in the snapshot. In most cases, this is the setting you want.

Independent disks are not included in the snapshot.

Caution: The independent disk option should be used only by advanced users who need it for special-purpose configurations.

You have the following options for an independent disk:

- **Persistent** — changes are immediately and permanently written to the disk.
- **Nonpersistent** — changes to the disk are discarded when you power off or reset the virtual machine.

When you have set the filename and location you want to use and have made any selections you want to make on the Specify Advanced Options screen, click **Finish**. GSX Server creates the virtual machine.

Your new virtual machine is like a physical computer with a blank hard disk. Before you can use it, you need to partition and format the virtual disk and install an operating system. The operating system's installation program may handle the partitioning and formatting steps for you. For information about installing the guest operating system, see the *VMware Guest Operating System Installation Guide*.

Creating a New Virtual Machine from the VMware Management Interface

You can create new virtual machines from the VMware Management Interface. The process sets up a new configuration for each virtual machine you create in this fashion. You do not need to use the New Virtual Machine Wizard in order to do this. Creating a virtual machine using the management interface is similar to following the typical path when creating a virtual machine with the New Virtual Machine Wizard, although the management interface configures the virtual machine with bridged networking. You can change the type of networking to network address translation (NAT) or host-only networking after you create the virtual machine.

The virtual machines you create are located on the host to which you are currently logged on, even if the browser you are using is running on a remote client.

As with any other virtual machine, you can change any configuration settings in the virtual machine settings editor in the console (**VM** > **Settings**); most settings can be configured in the management interface.

A virtual machine created with the management interface is private. If you want to make this virtual machine available to all users, change the setting in the virtual machine settings editor. For more information, see Only You Can See Virtual Machines You Create in the *VMware GSX Server Administration Guide*.

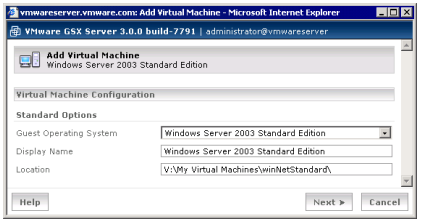
For more information about the management interface, see Using the VMware Management Interface in the *VMware GSX Server Administration Guide*.

Note: Before you create the virtual machine, you should check the installation notes for the guest operating system you intend to install in it. You can find this information

in the *VMware Guest Operating System Installation Guide* available from the Help menu or from the VMware Web site at *www.vmware.com/support/guestnotes/doc/index.html*.

To create a virtual machine from the VMware Management Interface, complete the following steps.

1. On the Status Monitor page, click **Add Virtual Machine**. The Add Virtual Machine page appears.



2. In the **Guest Operating System** list, select the guest operating system for the new virtual machine. A name for the virtual machine appears in the **Display Name** field; a default path to the configuration file appears in the **Location** field.

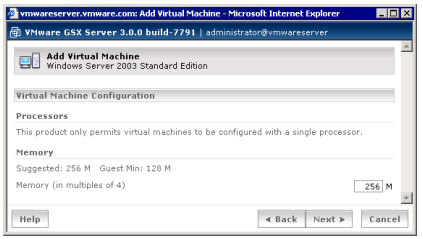
   Note: If you are creating a Red Hat Enterprise Linux 4 virtual machine, select **Other Linux 2.6.x Kernel**.

   If you want, you can change the display name for the new virtual machine. In the **Display Name** field, type a descriptive name of the new virtual machine. This name appears in the **Display Name** column in the management interface and in the VMware Virtual Machine Console.

   If you want, you can change the path to the new virtual machine. In the **Location** field, type the path to the new virtual machine's configuration file on the host machine.

   Each virtual machine must have its own directory. All associated files, such as the configuration file and the disk files, are placed in this directory.

   After you make your selections, click **Next** to continue.

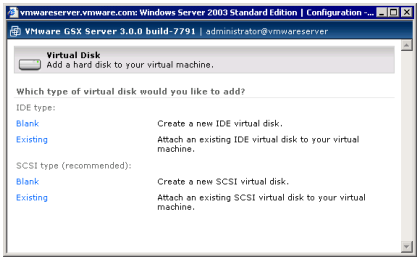

vmwareserver.vmware.com: Add Virtual Machine - Microsoft Internet Explorer
VMware GSX Server 3.0.0 build-7791 | administrator@vmwareserver
Add Virtual Machine
Windows Server 2003 Standard Edition
Virtual Machine Configuration
Processors
This product only permits virtual machines to be configured with a single processor.
Memory
Suggested: 256 M Guest Min: 128 M
Memory (in multiples of 4)
256 M
Help
◂ Back
Next ▸
Cancel

3. Allocate memory to the virtual machine. The default setting in the **Memory** field depends on the guest operating system you have selected. You may need to change it to meet the demands of applications you plan to run in the virtual machine. The amount of memory you specify must be a multiple of four. You may change this setting later.

   Caution: You cannot allocate more than 2000MB of memory to a virtual machine if it is stored on a file system that cannot support files larger than 2GB, such as FAT16. You will not be able to power on such a virtual machine. Further, you cannot allocate more than 2000MB of memory to a virtual machine if it is stored on a FAT32 file system, even though it does support files up to 4GB in size.

   For more information about memory, see Understanding Memory Usage in the *VMware GSX Server Administration Guide* and Allocating Memory to a Virtual Machine on page 340.

   After you make your selection, click **Next**. The Disk page appears.


vmwareserver.vmware.com: Windows Server 2003 Standard Edition | Configuration -...
VMware GSX Server 3.0.0 build-7791 | administrator@vmwareserver
Virtual Disk
Add a hard disk to your virtual machine.
Which type of virtual disk would you like to add?
IDE type:
Blank
Create a new IDE virtual disk.
Existing
Attach an existing IDE virtual disk to your virtual machine.
SCSI type (recommended):
Blank
Create a new SCSI virtual disk.
Existing
Attach an existing SCSI virtual disk to your virtual machine.

4. Choose whether you want to add a new virtual disk to the virtual machine or use an existing one.

   - To create a new virtual disk, see Creating a New Virtual Disk on page 53
   - To add an existing virtual disk, see Using an Existing Virtual Disk on page 55

   After you create or add the virtual disk, proceed with the next step.

5. After you finish configuring the virtual disk, click **Next**. GSX Server creates the virtual machine and preallocates all the virtual disk space. The new virtual machine appears on the Status Monitor page.

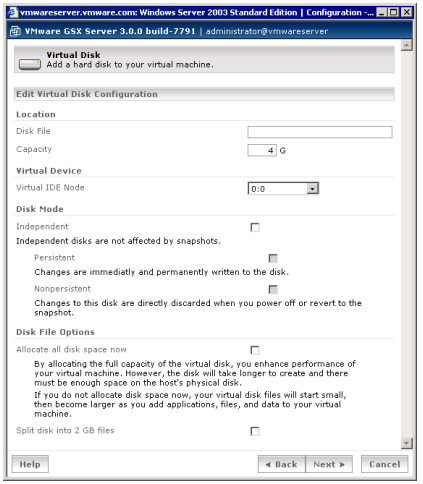
6. The Hardware tab for this virtual machine appears.

   You can change any of the default settings GSX Server assigned to the virtual machine (such as the network adapter and any removable devices) or configuration items you specified as you create the virtual machine. To change

any hardware, see Configuring a Virtual Machine's Hardware in the *VMware GSX Server Administration Guide*.

Creating a New Virtual Disk

1. To create a new virtual disk, decide whether you want the disk to be IDE or SCSI. The wizard suggests the recommended type. Then under **IDE type** or **SCSI type**, click **Blank**. The Virtual Disk Configuration page appears.


vmwareserver.vmware.com: Windows Server 2003 Standard Edition | Configuration -...
VMware GSX Server 3.0.0 build-7791 | administrator@vmwareserver
Virtual Disk
Add a hard disk to your virtual machine.
Edit Virtual Disk Configuration
Location
Disk File
Capacity
4 G
Virtual Device
Virtual IDE Node
0:0
Disk Mode
Independent
Independent disks are not affected by snapshots.
Persistent
Changes are immediatly and permanently written to the disk.
Nonpersistent
Changes to this disk are directly discarded when you power off or revert to the snapshot.
Disk File Options
Allocate all disk space now
By allocating the full capacity of the virtual disk, you enhance performance of your virtual machine. However, the disk will take longer to create and there must be enough space on the host's physical disk.
If you do not allocate disk space now, your virtual disk files will start small, then become larger as you add applications, files, and data to your virtual machine.
Split disk into 2 GB files
Help
Back
Next
Cancel

2. In the **Disk File** field, enter the location and name of the virtual disk. Make sure the virtual disk has a `.vmdk` extension.
3. In the **Capacity** field, specify the size of the virtual disk in Gigabytes (GB). By default, GSX Server preallocates the space for the virtual disk when you create it.

   The virtual disk can be as small as 0.1GB (100MB). A SCSI virtual disk can be as large as 256GB; an IDE virtual disk can be as large as 128GB. The default is 4GB.
4. Specify the virtual device node in the **Virtual IDE Node** or **Virtual SCSI Node** list as appropriate.
5. Decide if you want to make this virtual disk an independent disk. Under **Disk Mode**, check **Independent**, then check **Persistent** or **Nonpersistent**.

   Independent disks are not included in a snapshot.

   Caution: The independent disk option should be used only by advanced users who need it for special-purpose configurations.

You have two options for an independent disk. You can make the disk **Persistent**, which means that changes are immediately and permanently written to the disk. Or you can make the disk **Nonpersistent**, which means that changes to the disk are discarded when you power off or reset the virtual machine.

6. Decide if you want to preallocate the virtual disk space. Allocating all the space at the time you create the virtual disk gives somewhat better performance and ensures you do not run out of disk space on the host, but it requires as much disk space as the size you specify for the virtual disk. You cannot shrink a preallocated disk. To preallocate the virtual disk, check the **Allocate all disk space now** box.

   A preallocated virtual disk is needed for clustering virtual machines. For more information about clustering, see High-Availability Configurations with VMware GSX Server in the *VMware GSX Server Administration Guide*.

   If you do not preallocate the disk, the virtual disk's files start small and grow as needed.

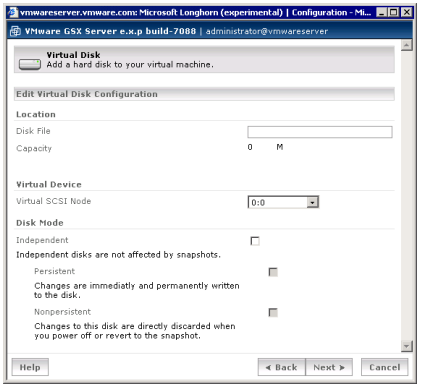
   You may also specify whether you want the virtual disk created as one large file or split into a set of 2GB files. You should split the virtual disk it is stored on a FAT32 file system or a file system that cannot support files larger than 2GB, such as FAT16. To do this, check **Split into 2GB files**.

   If you later decide you want to enlarge a virtual disk, change the virtual disk from preallocated to growable (or vice versa), or split a virtual disk that was created in a single file (or vice versa), you can use the VMware Virtual Disk Manager to expand or convert the virtual disk. For more information, see Using VMware Virtual Disk Manager on page 191.

7. Continue with step 5 under Creating a New Virtual Machine from the VMware Management Interface on page 50.

Using an Existing Virtual Disk

1. To add an existing virtual disk, decide whether the disk is IDE or SCSI. Then under **IDE type** or **SCSI type**, click **Blank**. The Virtual Disk Configuration page appears.



2. In the **Disk File** field, enter the location for the virtual disk.
3. Specify the virtual device node in the **Virtual IDE Node** or **Virtual SCSI Node** list as appropriate.
4. Decide if you want to make this virtual disk an independent disk. Under **Disk Mode**, check **Independent**, then check **Persistent** or **Nonpersistent**.

   Independent disks are not included in the snapshot.

   Caution: The independent disk option should be used only by advanced users who need it for special-purpose configurations.

   You have two options for an independent disk. You can make the disk **Persistent**, which means that changes are immediately and permanently written to the disk. Or you can make the disk **Nonpersistent**, which means that changes to the disk are discarded when you power off or reset the virtual machine.
5. Continue with step 5 under Creating a New Virtual Machine from the VMware Management Interface on page 50.

Installing a Guest Operating System

A new virtual machine is like a physical computer with a blank hard disk. Before you can use it, you need to partition and format the virtual disk and install an operating system. The operating system's installation program may handle the partitioning and formatting steps for you.

Installing a guest operating system inside your VMware GSX Server virtual machine is essentially the same as installing it on a physical computer. The basic steps for a typical operating system are:

1. Launch the VMware Virtual Machine Console.
2. Insert the installation CD-ROM or floppy disk for your guest operating system.

   Note: If you plan to use a PXE server to install the guest operating system over a network connection, you don't need the operating system installation media. When you power on the virtual machine in the next step, the virtual machine detects the PXE server, if one is available on the network. For more information, see Using PXE with Virtual Machines on page 132.

   Note: In some host configurations, the virtual machine is not able to boot from the installation CD-ROM. You can work around that problem by creating an ISO image file from the installation CD-ROM. Use the virtual machine settings editor (**VM** > **Settings**) to connect the virtual machine's CD-ROM drive to the ISO image file, then power on the virtual machine.
3. Power on your virtual machine by clicking the **Power On** button.
4. Follow the instructions provided by the operating system vendor.

For a brief illustration on installing a Windows Server 2003 guest operating system, see Example: Installing Windows Server 2003 as a Guest Operating System on page 56. The screen shots illustrate the process on a Windows host. The steps are the same on a Linux host.

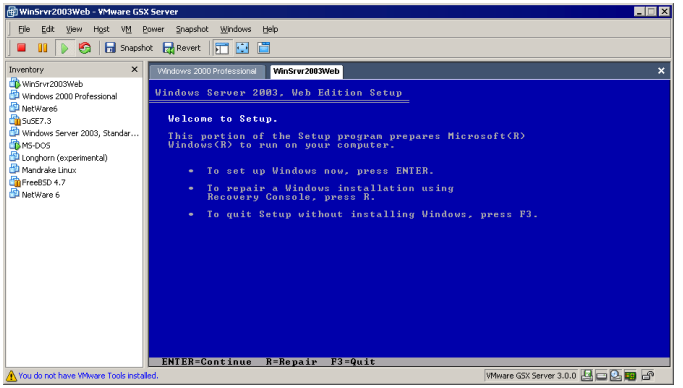
For information on installing other guest operating systems, see the *VMware Guest Operating System Installation Guide*, available from the Help menu or from the VMware Web site at *www.vmware.com/support/guestnotes/doc/index.html*.

Example: Installing Windows Server 2003 as a Guest Operating System

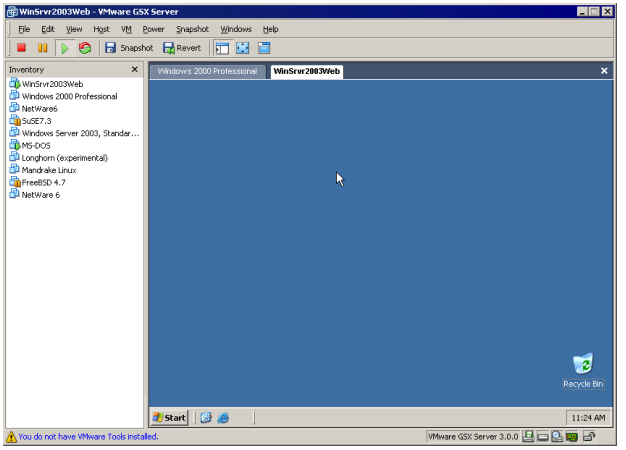
You can install Windows Server 2003 Standard Edition, Enterprise Edition or Web Edition in a virtual machine using the corresponding Windows Server 2003 distribution CD.

Installation Steps

1. Insert the Windows Server 2003 CD in the CD-ROM drive.
2. Power on the virtual machine to start installing Windows Server 2003.


WinSrvr2003Web - VMware GSX Server
File Edit View Host VM Power Snapshot Windows Help
Snapshot Revert
Inventory
WinSrvr2003Web
Windows 2000 Professional
NetWare6
SuSE7.3
Windows Server 2003, Standar...
MS-DOS
Longhorn (experimental)
Mandrake Linux
FreeBSD 4.7
NetWare 6
Windows 2000 Professional
WinSrvr2003Web
Windows Server 2003, Web Edition Setup
Welcome to Setup.
This portion of the Setup program prepares Microsoft(R) Windows(R) to run on your computer.
To set up Windows now, press ENTER.
To repair a Windows installation using Recovery Console, press R.
To quit Setup without installing Windows, press F3.
ENTER=Continue R=Repair F3=Quit
You do not have VMware Tools installed.
VMware GSX Server 3.0.0

3. If you enabled the virtual machine's Ethernet adapter, an AMD PCNET Family Ethernet Adapter is detected and set up automatically.
4. Follow the installation steps as you would for a physical computer.



After installing your guest operating system, you are ready to install VMware Tools as described in Installing VMware Tools on page 62.

For more information about using Windows Server 2003 guest operating systems, such as enabling networking in the virtual machine, see the *VMware Guest Operating System Installation Guide*, available from the VMware Web site or from the Help menu.

CHAPTER 3

Using VMware Tools

The following sections describe how to install and run VMware Tools:

- About VMware Tools on page 60
- Installing VMware Tools on page 62
- Executing Scripts When the Virtual Machine's Power State Changes on page 76
- Configuring VMware Tools on page 78
- About the VMware Tools Service on page 90

About VMware Tools

VMware Tools is a suite of utilities that enhances the performance of the virtual machine's guest operating system and improves management of the virtual machine by VMware GSX Server. It is very important that you install VMware Tools in the guest operating system. Although GSX Server can run a guest operating system without VMware Tools, you lose important functionality and convenience.

When you install VMware Tools, you install •The VMware Tools service (or `vmware-guestd` on Linux guests).

- A set of VMware device drivers, including an SVGA display driver, the `vmxnet` networking driver for some guest operating systems, the BusLogic SCSI driver for some guest operating systems and the VMware mouse driver.
- The VMware Tools control panel that lets you modify settings, shrink virtual disks, and connect and disconnect virtual devices.
- A set of scripts that help automate guest operating system operations; the scripts run when the virtual machine's power state changes.
- A component that supports copying and pasting text between the guest and host operating systems.

VMware Tools performs various duties within the guest operating system, such as passing messages from the host operating system to the guest operating system, sending a heartbeat to GSX Server, grabbing and releasing the mouse cursor, and synchronizing the time in the guest operating system with the time in the host operating system. The service starts automatically when the guest operating system boots. For more information, see About the VMware Tools Service on page 90.

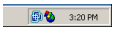
With the VMware SVGA driver installed, GSX Server supports up to 32-bit displays and high display resolution, with significantly faster overall graphics performance. If you run a guest operating system without VMware Tools, the graphics environment within the virtual machine is limited to VGA mode graphics (640x480, 16 color) and display performance may be unsatisfactory.

The VMware virtual SCSI driver is a BusLogic driver. Note that some recent guest operating systems contain LSI Logic drivers and can take advantage of the virtual LSI Logic adapter for better device performance.

The `vmxnet` networking driver improves network performance.

The VMware mouse driver improves mouse performance in some guest operating systems. It is necessary for use with third party tools like Microsoft's Terminal Services.

In a Windows guest, you can access the VMware Tools control panel through the Windows Control Panel (choose **Start** > **Settings** > **Control Panel** > **VMware Tools**) or via the VMware Tools icon, which appears by default in the system tray.



In a Linux or FreeBSD guest operating system, the VMware Tools control panel is called `vmware-toolbox`. You can launch it manually as a background process from a terminal using

```
vmware-toolbox &
```

Note: To get the greatest benefit from the features of VMware Tools, always run `vmware-toolbox` in the guest operating system.

In a NetWare 5.1 or higher guest operating system, you can access the VMware Tools control panel by choosing **Novell** > **Settings** > **VMware Tools for NetWare**.

In a NetWare 4.2 guest operating system, you can use VMware Tools commands in the system console. The VMware Tools program is called `vmwtool`. For information about using this command, see Configuring VMware Tools for NetWare Guests in the System Console on page 87.

With some window managers, you can place the command to start VMware Tools in a startup configuration so VMware Tools starts automatically when you start your graphical environment. Consult your window manager's documentation for details.

Installation files for VMware Tools for all supported Windows, Linux, NetWare and FreeBSD guest operating systems are built into GSX Server.

Note: There is no VMware Tools package for Solaris guest operating systems.

Installing VMware Tools

The following sections describe how to install VMware Tools:

- Installing VMware Tools in a Windows Virtual Machine on page 62
  - Automating the Installation of VMware Tools in a Windows Guest on page 69
- Additional Steps for Some Versions of Windows When Migrating from Old Disk Versions on page 66
- Installing VMware Tools in a Linux or FreeBSD Virtual Machine on page 71
- Installing VMware Tools in a NetWare Virtual Machine on page 74

The installers for VMware Tools for Windows, Linux, FreeBSD and NetWare guest operating systems are built into VMware GSX Server as ISO image files. (An ISO image file looks like a CD-ROM to your guest operating system and even appears as a CD-ROM in Windows Explorer. You do not use an actual CD-ROM to install VMware Tools, nor do you need to download the CD-ROM image or burn a physical CD-ROM of this image file.)

When you install VMware Tools, GSX Server temporarily connects the virtual machine's first virtual CD-ROM drive to the ISO image file that contains the VMware Tools installer for your guest operating system, then begins the installation process. (If you decide not to proceed with the installation, cancel the installer, then choose **VM** > **Cancel VMware Tools Install** to return your virtual machine's CD-ROM drive to its original configuration.)

Installing VMware Tools in a Windows Virtual Machine

VMware Tools for Windows guest operating systems supports all Windows guest operating systems.

The detailed steps for installing VMware Tools depend on the version of Windows you are running. The steps that follow show how to install VMware Tools in a Windows Server 2003 guest. Some steps that are automated in current versions of Windows must be performed manually in Windows 9x and Windows NT.

Note: If you are running GSX Server on a Windows host and your virtual machine has only one CD-ROM drive, the CD-ROM drive must be configured as an IDE or SCSI CD-ROM drive. It cannot be configured as a generic SCSI device.

To add an IDE or SCSI CD-ROM drive, see Adding, Configuring and Removing Devices in a Virtual Machine on page 141. For information about generic SCSI, see Connecting to a Generic SCSI Device on page 325.

You can automate the installation of VMware Tools in a Windows guest operating system. For information, see Automating the Installation of VMware Tools in a Windows Guest on page 69.

Installing VMware Tools in a Windows Guest Operating System

1. Power on the virtual machine.
2. When the guest operating system starts, prepare your virtual machine to install VMware Tools.

   Choose **VM > Install VMware Tools**.

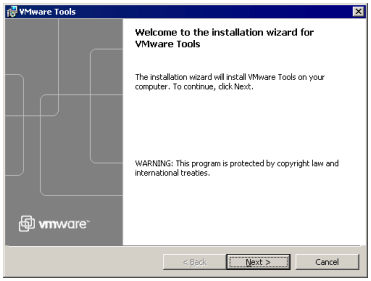
   The remaining steps take place inside the virtual machine.
3. Log on to the virtual machine as an administrator.

   Note: You must be an administrator to install VMware Tools in a Windows guest operating system, unless the guest operating system is Windows Me, Windows 98 or other early versions of Windows.
4. If you have autorun enabled in your guest operating system (the default setting for Windows operating systems), a dialog box appears after a few seconds. It asks if you want to install VMware Tools. Click **Yes** to launch the InstallShield wizard.

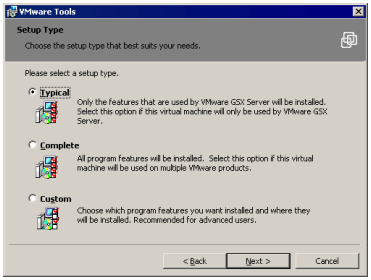
   If autorun is not enabled, the dialog box does not appear automatically. If it doesn't appear, run the VMware Tools installer. Click **Start** > **Run** and enter `D:\setup\setup.exe` where `D:` is your first virtual CD-ROM drive.

   Note: You do not use an actual CD-ROM to install VMware Tools, nor do you need to download the CD-ROM image or burn a physical CD-ROM of this image file. The VMware GSX Server software contains an ISO image that looks like a CD-ROM to your guest operating system and even appears as a CD-ROM in Windows Explorer. This image contains all the files needed to install VMware Tools in your guest operating system. When you finish installing VMware Tools, this image file no longer appears in your CD-ROM drive.

The VMware Tools installation wizard starts.


VMware Tools
Welcome to the installation wizard for VMware Tools
The installation wizard will install VMware Tools on your computer. To continue, click Next.
WARNING: This program is protected by copyright law and international treaties.
vmware
< Back
Next >
Cancel

5. Click **Next** to continue with the VMware Tools installation wizard. The Setup Type dialog box appears.


VMware Tools
Setup Type
Choose the setup type that best suits your needs.
Please select a setup type.
Typical
Only the features that are used by VMware GSX Server will be installed. Select this option if this virtual machine will only be used by VMware GSX Server.
Complete
All program features will be installed. Select this option if this virtual machine will be used on multiple VMware products.
Custom
Choose which program features you want installed and where they will be installed. Recommended for advanced users.
< Back
Next >
Cancel

6. Choose whether you want to perform a typical, complete or custom installation. The installer uses this selection each time you upgrade VMware Tools.

Typical Installation

A typical installation installs only those components used by GSX Server. For example, the driver to use shared folders (a feature in VMware Workstation) is not installed in a typical installation in a virtual machine created with GSX Server.

A typical installation installs the utilities to enhance the performance of the guest operating system, and a set of drivers specific to GSX Server virtual machines — the VMware SVGA driver, the VMware Mouse driver, the VMware SCSI driver and the VMware `vmxnet` networking driver (the `vlance` driver is installed automatically when you created the virtual machine).

If you plan on not using this virtual machine with other VMware products, such as VMware Workstation, you can use the typical installation. To choose the typical installation, select **Typical**, click **Next**, then go to step 7.

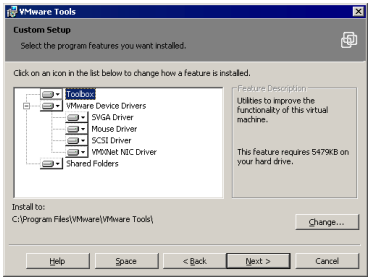
Complete Installation

A complete installation installs the utilities to enhance the performance of the guest operating system, and all the drivers — the VMware SVGA driver, the VMware Mouse driver, the VMware SCSI driver, the VMware `vmxnet` networking driver (the `vlance` driver is installed automatically when you created the virtual machine) and the shared folders driver (for use by virtual machines with VMware Workstation).

If you plan on using this virtual machine with other VMware products, use the complete installation. To choose the complete installation, select **Complete**, click **Next**, then go to step 7.

Custom Installation

A custom installation lets you pick and choose which components to install. You can always run the installer again at a later date to install components you did not install the first time, or remove components you no longer want. Select **Custom** and click **Next**. The Custom Setup screen appears.



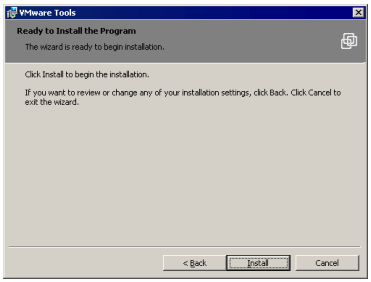
In the Custom Setup screen, pick and choose the components to install. Click the arrow to the left of the component you do not want to install and select the appropriate option from the menu.

If you need to determine how much free space is on the guest click **Space**. This is useful if you are choosing a custom installation due to limited disk space on your guest.

If you want to install all the VMware Tools components in a directory other than the default, click **Browse** and select the directory. If the directory does not exist, the installer creates it for you.

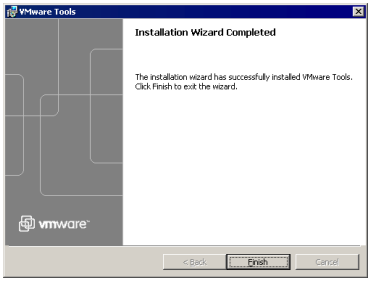
When you are ready to continue, click **Next**.

7. If you want to change any settings or information you provided, now is the time to make those changes. Click **Back** until you reach the dialog box containing the information you want to change.


VMware Tools
Ready to Install the Program
The wizard is ready to begin installation.
Click Install to begin the installation.
If you want to review or change any of your installation settings, click Back. Click Cancel to exit the wizard.
< Back
Install
Cancel

   Otherwise, click **Install**. The installer begins copying files to your host.

8. You may see one or more Digital Signature Not Found dialog boxes when the installer begins to install the virtual drivers. You can safely ignore these warnings and click **Yes** or **Continue** to approve installation of the drivers.



9. After the installer finishes installing the files, click **Finish**. If you installed the VMware SVGA driver, most Windows guest operating systems can use it only after you reboot the guest. With Windows XP guests, you do not have to reboot to use the new driver.

With some older Windows guest operating systems, extra steps are needed.

Additional Steps for Some Versions of Windows When Migrating from Old Disk Versions

If you are migrating a GSX Server 2 disk to GSX Server 3 and your guest operating system is Windows NT, Windows Me, Windows 98 or Windows 95, you need to configure the video driver by hand. Instructions open automatically in Notepad at the end of the installation process. If the Notepad window is hidden, bring it to the front by clicking the **Notepad** button on the Windows taskbar.

For details, see the steps below that correspond to your guest operating system.

Windows NT

1. After installing VMware Tools, click **Finish**. The Display Properties dialog box appears.
2. Click the **Display Type** button. The Display Type dialog box appears.
3. Click the **Change** button. The Change Display dialog box appears.
4. Select **VMware, Inc.** from the **Manufacturer** list.
5. Select **VMware SVGA** as the display adapter and click **OK**.
6. Click **Yes** in response to the on-screen question about third-party drivers to install the driver, then click **OK** to confirm the drivers were installed.
7. Click **Close** from the Display Type dialog box, then click **Close** from the Display Properties dialog box.
8. Click **Yes** to restart Windows NT and start using the new video driver.
9. The VMware Tools background application is launched automatically when you reboot your virtual machine.

Windows Me

1. After installing VMware Tools, click **Finish**. The Display Settings dialog box appears.
2. Click the **Advanced** button.
3. Click the **Adapter** tab.
4. Click the **Change** button. This starts the Update Device Driver wizard.
5. The wizard now presents two options. Choose the second option to **Specify the location of the driver**.

   Click **Next**.
6. Check the **Specify a location** checkbox. Enter the following path:

   `D:\video\win9x`

   `D:` is the drive letter for the first virtual CD-ROM drive in your virtual machine.

   Click **OK**.
7. Windows Me automatically locates your driver.
8. Select the **VMware SVGA II** display adapter and click **Next**.
9. Click **Next** to install the driver.

If you are upgrading a virtual machine created under GSX Server 2, you may see a dialog box that warns, "The driver you are installing is not specifically designed for the hardware you have.… Do you wish to continue?" Click **Yes**.

After the driver is installed, click **Finish**.

10. Click **Yes** to restart Windows Me and start using the new video driver.
11. The VMware Tools background application starts automatically when you reboot your virtual machine.

Windows 98

1. After installing VMware Tools, click **Finish**. The Display Settings dialog box appears.
2. Click the **Advanced** button. The Standard Display Adapter (VGA) Properties dialog box appears. If you are upgrading from a previous version of the VMware drivers, this dialog box is titled VMware SVGA Properties.
3. Click the **Adapter** tab.
4. Click the **Change** button. This starts the Update Device Driver wizard. Click **Next**.
5. The wizard presents two options. Choose the option to **Display a list of all drivers in a specific location**. Click **Next**.
6. Select **Have Disk**. The Install From Disk dialog box appears.
7. Enter the following path:

   `D:\video\win9x`

   `D:` is the drive letter for the first virtual CD-ROM drive in your virtual machine.

   Click **OK**.
8. Select **VMware SVGA** display adapter and click **OK**.
9. Answer **Yes** to the on-screen question, then click **Next** to install the driver. After the driver is installed, click **Finish**.
10. Click **Close** in the SVGA Properties dialog box, then click **Close** in the Display Settings dialog box.
11. Click **Yes** to restart Windows 98 and start using the new video driver.
12. The VMware Tools background application starts automatically when you reboot your virtual machine.

Windows 95

1. After installing VMware Tools, click **Finish**. The Display Settings dialog box appears.
2. Click the **Advanced Properties** button. The Advanced Display Properties dialog box appears.
3. Click the **Change** button. The Select Device dialog box appears.
4. Select **Have Disk**.
5. Enter the following path:

   `D:\video\win9x`

   `D:` is the drive letter for the first virtual CD-ROM drive in your virtual machine.

   Click **OK**.
6. Click **OK** again to install the driver.
7. Click **Close** from the Advanced Display Properties dialog box, then click **Close** from the Display Setting dialog box.
8. Click **Yes** to restart Windows 95 and start using the new video driver.
9. The VMware Tools background application starts automatically when you reboot your virtual machine.

Automating the Installation of VMware Tools in a Windows Guest

To automate the installation of VMware Tools in a Windows guest operating system, you can use the Microsoft Windows Installer runtime engine to install the software silently (in quiet mode). If you are installing VMware Tools in a number of Windows virtual machines, you may want to use the silent install features.

The guest operating system in which you are installing VMware Tools must have Microsoft Windows Installer runtime engine version 2.0 or higher installed. This version is included with Windows Server 2003 and Windows XP. If you are installing VMware Tools in other Windows guest operating systems, check the version of this file:

```
%WINDIR%\system32\msiexec.exe
```

If you need to upgrade the engine, run `instmsiw.exe` (`instmsia.exe` for Windows 95 or Windows 98 guests), which is included with the VMware Tools installer.

For more information on using the Microsoft Windows Installer, go to the Microsoft Web site — *msdn.microsoft.com/library/default.asp?url=/library/en-us/msi/setup/about_windows_installer.asp*.

To install VMware Tools silently in a Windows guest, first make sure the virtual machine's CD-ROM drive is connected to the VMware Tools ISO image (`windows.iso`, located in the directory where you installed GSX Server) and configured to connect when you power on the virtual machine. Then, run the silent installation on the extracted installation packages. At the command prompt, on one line, type:

```
msiexec -i "D:\VMware Tools.msi" ADDLOCAL=ALL /qn
```

The installation command can be customized using standard Microsoft Windows Installer installation options.

The `ADDLOCAL` option defaults to install all VMware Tools components. You can customize the installation using a combination of the `ADDLOCAL` and `REMOVE` options. For information about the features of VMware Tools, see About VMware Tools on page 60. You can include or exclude the following features:

- `Toolbox` — the VMware Tools control panel and its utilities. Excluding this feature prevents you from using VMware Tools in the guest operating system, and is not recommended.
- `Drivers` — this includes the SVGA, Mouse, BusLogic and `vmxnet` drivers.
  - `SVGA` — the VMware SVGA driver. Excluding this feature limits the display capabilities of your virtual machine.
  - `Mouse` — the VMware mouse driver. Excluding this feature decreases mouse performance in your virtual machine.
  - `Buslogic` — the VMware BusLogic driver. Excluding this feature prevents you from using this driver in your virtual machine. If your virtual machine is configured to use the LSI Logic driver, then you may want to remove this feature.
  - `VMXNet` — the VMware `vmxnet` networking driver. Excluding this feature prevents you from using this driver in your virtual machine.
  - `MemCtl` — the VMware memory control driver. This feature is recommended if you plan on using this virtual machine with VMware ESX Server. Excluding this feature hinders the memory management capabilities of the virtual machine running on an ESX Server system.
- `Hgfs` — the VMware shared folders driver. This feature is recommended if you plan on using this virtual machine with VMware Workstation. Excluding this feature prevents you from sharing a folder between your virtual machine and the Workstation host.

To include a feature, use it with the `ADDLOCAL` option.

To exclude a feature, use it with the `REMOVE` option.

For example, to install everything but the shared folders driver, type the following on the command line:

```
msiexec -i "D:\VMware Tools.msi" ADDLOCAL=ALL REMOVE=Hgfs /qn
```

The `SVGA`, `Mouse`, `BusLogic`, `vmxnet` and `MemCtl` features are children of the `Drivers` feature. Thus, on the command line, if you type

```
msiexec -i "D:\VMware Tools.msi" ADDLOCAL=ALL
REMOVE=Drivers /qn
```

you also skip installation of the `SVGA`, `Mouse`, `BusLogic`, `vmxnet` and `MemCtl` drivers.

The drivers installed by VMware Tools are not signed by Microsoft. When you install VMware Tools, you are asked to confirm the installation of these drivers. You can prevent these messages from appearing in the guest operating system during installation by completing the following steps.

1. On the virtual machine's desktop, right-click **My Computer**, then choose **Properties**.
2. Click the **Hardware** tab, then click **Driver Signing**. The Driver Signing dialog box appears.
3. Click **Ignore**, then click **OK** twice.

Installing VMware Tools in a Linux or FreeBSD Virtual Machine

1. Power on the virtual machine.
2. After the guest operating system has started, prepare your virtual machine to install VMware Tools.

   Choose **VM > Install VMware Tools**.

   The remaining steps take place inside the virtual machine.
3. Be sure the guest operating system is running in text mode. You cannot install VMware Tools from a terminal in an X window session.

   Some recent distributions of Linux are configured to run the X server when they boot and do not provide an easy way to stop the X server. However, you can switch to a different workspace that is still in text mode and install VMware Tools from that workspace.

To switch between Linux workspaces in a virtual machine, press Ctrl-Alt-Space, release Space without releasing Ctrl and Alt, then press the function key for the workspace you want to use — for example, F2.

Note: If you changed your hot-key combination to something other than Ctrl-Alt, use that combination with Space and the function key.

4. As root (`su -`), mount the VMware Tools virtual CD-ROM image, change to a working directory (for example, `/tmp`), uncompress the installer, then unmount the CD-ROM image.

   Note: You do not use an actual CD-ROM to install VMware Tools, nor do you need to download the CD-ROM image or burn a physical CD-ROM of this image file. The GSX Server software contains an ISO image that looks like a CD-ROM to your guest operating system. This image contains all the files needed to install VMware Tools in your guest operating system.

   Linux Guests: Some Linux distributions use different device names or organize the `/dev` directory differently. If your CD-ROM drive is not `/dev/cdrom` or if the mount point for a CD-ROM is not `/mnt/cdrom`, modify the following commands to reflect the conventions used by your distribution.

   Further, some Linux distributions automatically mount CD-ROMs. If your distribution uses automounting, do not use the `mount` and `umount` commands below. You still must untar the VMware Tools installer to `/tmp`.

   ```
   mount /dev/cdrom /mnt/cdrom
   cd /tmp
   tar zxf /mnt/cdrom/vmware-linux-tools.tar.gz
   umount /mnt/cdrom
   ```

   FreeBSD Guests: Some FreeBSD distributions automatically mount CD-ROMs. If your distribution uses automounting, do not use the `mount` and `umount` commands below. You still must untar the VMware Tools installer to `/tmp`.

   ```
   mount /cdrom
   cd /tmp
   tar zxf /cdrom/vmware-freebsd-tools.tar.gz
   umount /cdrom
   ```

5. Run the VMware Tools installer.

   ```
   cd vmware-tools-distrib
   ./vmware-install.pl
   ```

6. Answer the questions about default directories.

7. Run the configuration program.
   `vmware-config-tools.pl`
8. To change your virtual machine's display resolution, answer yes, then enter the number that corresponds to the desired resolution.
9. Log off of the root account.
   `exit`
10. Start X and your graphical environment.
11. In an X terminal, launch the VMware Tools background application.
    `vmware-toolbox &`

You can run VMware Tools as root or as a normal user. To shrink virtual disks, you must run VMware Tools as root (`su -`).

Note: To get the greatest benefit from the features of VMware Tools, always run `vmware-toolbox` in the guest operating system.

Starting VMware Tools Automatically

You may find it helpful to configure your guest operating system so VMware Tools starts when you start your X server. The steps for doing so vary depending on your Linux distribution and your desktop environment. Check your operating system documentation for the appropriate steps to take.

For example, in a Red Hat Linux 7.1 guest using GNOME, follow these steps.

1. Open the Startup Programs screen in the GNOME Control Center.
   Main Menu (click the foot icon in the lower left corner of the screen) > Programs > Settings > Session > Startup Programs
2. Click Add.
3. In the Startup Command field, enter `vmware-toolbox`.
4. Click OK, click OK again, then close the GNOME Control Center.

The next time you start X, VMware Tools starts automatically.

Starting VMware Tools in a FreeBSD 4.5 Guest Operating System

In a FreeBSD 4.5 guest operating system, sometimes VMware Tools does not start after you install VMware Tools, reboot the guest operating system or start VMware Tools on the command line in the guest. An error message appears:

```
Shared object 'libc.so.3' not found.
```

The required library was not installed. This does not happen with full installations of FreeBSD 4.5, but does occur for minimal installations. To fix the problem of the missing library, take the following steps:

1. Insert and mount the FreeBSD 4.5 installation CD or access the ISO image file.
2. Change directories and run the installation script.

```
cd /cdrom/compat3x
./install.sh
```

Uninstalling VMware Tools

If you need to remove VMware Tools from your Linux guest operating system, log on as root (`su -`) and run the following command:

```
vmware-uninstall-tools.pl
```

Installing VMware Tools in a NetWare Virtual Machine

VMware Tools is available for NetWare 4.2, 5.1, 6.0 and 6.5 guest operating systems.

When you install VMware Tools in a NetWare guest operating system, the CPU idler program is installed and loaded automatically. The idler can be disabled from the system console. For information on configuring VMware Tools from the system console, see Configuring VMware Tools for NetWare Guests in the System Console on page 87.

Follow the appropriate steps for your NetWare guest operating system.

Installing VMware Tools in a NetWare 5.1, 6.0 or 6.5 Virtual Machine

1. Power on the virtual machine.
2. Prepare your virtual machine to install VMware Tools.

   Choose **VM > Install VMware Tools**.

   The remaining steps take place inside the virtual machine.
3. Load the CD-ROM driver so the CD-ROM device mounts the ISO image as a volume. Do one of the following.
   - In the system console for a NetWare 6.5 virtual machine, type `LOAD CDDVD`
   - In the system console for a NetWare 6.0 or NetWare 5.1 virtual machine, type `LOAD CD9660.NSS`
4. When the driver finishes loading, you can begin installing VMware Tools. In the system console, type `vmwtools:\setup.ncf`

When the installation finishes, the message `VMware Tools for NetWare are now running` appears in the Logger Screen (NetWare 6.5 and NetWare 6.0 guests) or the Console Screen (NetWare 5.1 guests).

5. Restart the guest operating system. In the system console, type
   `restart server`

After you install VMware Tools, make sure the VMware Tools virtual CD-ROM image (`netware.iso`) is not attached to the virtual machine. If it is, disconnect it. Right-click the CD-ROM icon in the status bar of the console window and select **Disconnect**.

Installing VMware Tools in a NetWare 4.2 Virtual Machine

1. Power on the virtual machine.
2. Prepare your virtual machine to install VMware Tools. Choose **VM > Install VMware Tools**. The remaining steps take place inside the virtual machine.
3. Load the `cdrom.nlm` module. In the system console, type
   `load cdrom`
4. Mount the VMware Tools CD-ROM image. In the system console, type
   `cd mount vmwtools`
5. Start installing VMware Tools. In the system console, type
   `vmwtools:\setup`

   When the installation finishes, the message `VMware Tools for NetWare are now running` appears in the Console Screen.
6. Bring the guest operating system down. In the system console, type
   `down`
7. Restart the guest operating system. In the system console, type
   `restart server`

After you install VMware Tools, make sure the VMware Tools virtual CD-ROM image (`netware.iso`) is not attached to the virtual machine. If it is, disconnect it. Right-click the CD-ROM icon in the status bar of the console window and select **Disconnect**.

Executing Scripts When the Virtual Machine's Power State Changes

You can run scripts in the guest operating system when you change the power state of a virtual machine; that is, when you power on, power off, suspend or resume the virtual machine.

Scripts can help automate guest operating system operations when you change the virtual machine's power state.

You perform these power operations from the toolbar buttons and menus in the VMware Virtual Machine Console and the VMware Management Interface.

You can configure scripts to run automatically when you use the power buttons on the toolbar by choosing **VM** > **Settings** > **Options** > **Power**, then checking the appropriate options under **Run VMware Tools scripts**.

Note: The commands on the **Power** menu take precedence over how the toolbar power buttons are configured.

Scripts can be executed only when the VMware Tools service is running. The service is a part of VMware Tools, so VMware Tools must be running in the guest in order for scripts to run. The service starts by default when you start the guest operating system. For more information about the VMware Tools service, see About the VMware Tools Service on page 90.

Default scripts are included in VMware Tools. On a Windows host, the default script executed when you suspend a virtual machine releases the IP address of the virtual machine, while the default script executed when you resume a virtual machine renews the IP address of the virtual machine (this affects only virtual machines configured to use DHCP). On a Linux host, the default script executed when you suspend a virtual machine stops networking for the virtual machine, while the default script executed when you resume a virtual machine starts networking for the virtual machine.

In addition, you can create your own scripts. The scripts you can run must be batch files for Windows hosts, but can be any executable format (such as shell or Perl scripts) for Linux hosts. You should have a thorough familiarity with these types of scripts before you modify the default scripts or create your own.

If you create your own scripts, you must associate each script with its particular power operation. For more information, see Choosing Scripts for VMware Tools to Run

During Power State Changes on page 79 for Windows guests and Choosing Scripts for VMware Tools to Run During Power State Changes on page 83 for Linux guests.

In order for scripts and their associated power operations to work, the following conditions must be met:

1. The VMware Tools service must be running in the virtual machine.
2. The version of VMware Tools must be updated to the current version. If you are using a virtual machine created with an older version of GSX Server or another older VMware product, update VMware Tools to the version included in this release.
3. Depending upon the operation the script performs, the virtual machine must have a virtual network adapter connected, otherwise the power operation fails.

Issues to Consider

Caution: When you reinstall VMware Tools after you upgrade the GSX Server software, any changes you made to the default scripts are overwritten. Any scripts you created on your own remain untouched, but do not benefit from any underlying changes that enhance the default scripts.

Note: Scripts cannot be run in NetWare, FreeBSD and Windows 95 guest operating systems.

Configuring VMware Tools

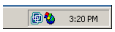
The following sections describe how to configure VMware Tools in a virtual machine.

- Configuring VMware Tools in a Windows Virtual Machine on page 78
- Configuring VMware Tools in a Linux or FreeBSD Virtual Machine on page 82
- Configuring VMware Tools in a NetWare Virtual Machine on page 85

Configuring VMware Tools in a Windows Virtual Machine

This section shows the options available in a Windows 2000 guest operating system. Similar configuration options are available in VMware Tools for other Windows guests.

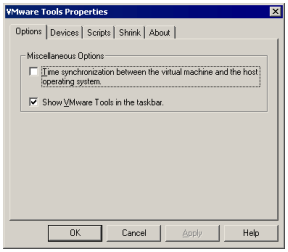
To open the VMware Tools control panel, double-click the VMware Tools icon in the system tray.



If the VMware Tools icon does not appear in the system tray, go to **Start** > **Control Panel** > **VMware Tools**.

Setting Options with VMware Tools

The Options tab shows miscellaneous options.



- **Time synchronization between the virtual machine and the host operating system** — this option lets you synchronize the time in the guest operating system with the time in the host operating system.

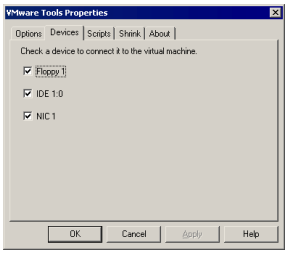
  Note: You can synchronize the time in the guest operating system with the time on the host operating system only when you set the clock in the guest operating system to a time earlier than the time set in the host.

  To completely disable time synchronization, see Disabling Time Synchronization on page 91.

- **Show VMware Tools in the taskbar** — this option displays the VMware Tools icon in the Windows taskbar.

Connecting Devices with VMware Tools

The Devices tab allows you to enable or disable removable devices. Removable devices include the floppy and CD-ROM drives and the virtual network adapter.



To connect a device, check the check box next to the device. To disconnect the device, clear the check box next to the device.

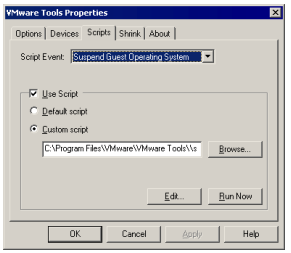
Note: You can also set these options from the **VM** > **Removable Devices** menu in the virtual machine window.

Choosing Scripts for VMware Tools to Run During Power State Changes

Through VMware Tools, you can run scripts that execute when you power on, power off, suspend or resume the virtual machine. For more information, see Executing Scripts When the Virtual Machine's Power State Changes on page 76.

Note: Scripts cannot be run in Windows 95 guest operating systems.

Note: Scripts in Windows NT and Windows Me guest operating systems do not release and renew the IP address.



The Scripts tab lets you enable, disable and run scripts that are associated with the Suspend, Resume, Power On and Power Off buttons.

A default script for each power state is included in VMware Tools. These scripts are located in the guest operating system in `C:\Program Files\VMware`.

When You …	This Default Script Runs
Suspend the guest operating system	`suspend-vm-default.bat`
Resume the guest operating system	`resume-vm-default.bat`
Power off the guest operating system	`poweroff-vm-default.bat`
Power on the guest operating system	`poweron-vm-default.bat`

Windows hosts: If the virtual machine is configured to use DHCP, the script executed when you suspend a virtual machine releases the IP address of the virtual machine. The script executed when you resume a virtual machine renews the IP address of the virtual machine.

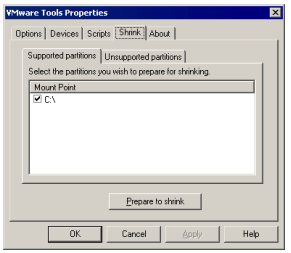
Linux hosts: The script executed when you suspend a virtual machine stops networking for the virtual machine. The script executed when you resume a virtual machine starts networking for the virtual machine.

For each power state, you can use the default script or you can substitute a script you created. In addition, you can test a script or disable the running of a script. Complete the following steps.

1. In the **Script Event** list, select the power operation with which to associate the script.
2. Do one of the following:
   - To select a different script, click **Custom Script**, then click **Browse** and select the new script.
   - To edit a script, click **Edit**. The script opens in your default editor. Make your changes there.
   - To test the script, click **Run Now**.
   - To disable the running of a script, clear the **Use Script** check box.
3. Click **Apply** to save your settings.

Shrinking Virtual Disks with VMware Tools

The Shrink tab gives you access to the controls you need if you wish to reclaim unused space in a virtual disk.

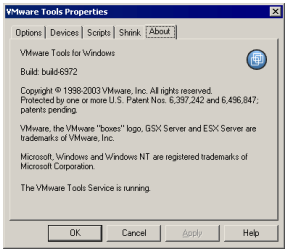


In some configurations, it is not possible to shrink virtual disks. If your virtual machine uses such a configuration, the Shrink tab displays information explaining why you cannot shrink your virtual disks.

For more information about shrinking virtual disks, see Defragmenting and Shrinking Virtual Disks on page 168.

Viewing Information About VMware Tools

For general information about VMware Tools, click the **About** tab.



In addition to copyright information, this tab contains the following information:

- The VMware Tools build number, which lets you verify your VMware Tools version matches the GSX Server version you are running. The build number is also useful when you request support.
- An indication as to whether or not the VMware Tools service is running.

Configuring VMware Tools in a Linux or FreeBSD Virtual Machine

This section shows the options available in a Linux or FreeBSD guest operating system.

To open the VMware Tools control panel, at a command prompt, type:

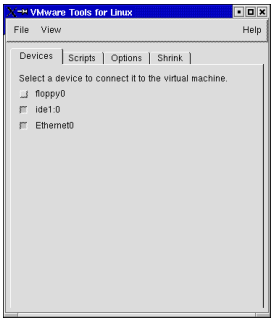
```
vmware-toolbox &
```

You may run VMware Tools as root or as a normal user. To shrink virtual disks, you should run VMware Tools as root (`su -`).

Note: To get the greatest benefit from the features of VMware Tools, always run `vmware-toolbox` in the guest operating system.

Connecting Devices with VMware Tools

The Devices tab allows you to enable or disable removable devices. Removable devices include the floppy and CD-ROM drives and the virtual network adapter.



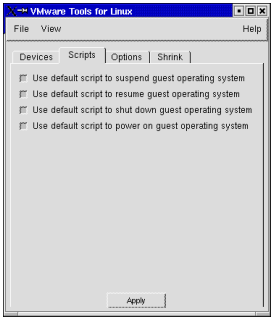
To connect a device, select the check box next to the device. To disconnect the device, deselect the check box next to the device.

Note: You can also set these options from the **VM** > **Removable Devices** menu in the virtual machine window.

Choosing Scripts for VMware Tools to Run During Power State Changes

Through VMware Tools, you can run scripts that execute when you power on, power off, suspend or resume the virtual machine. For more information, see Executing Scripts When the Virtual Machine's Power State Changes on page 76.

Note: Scripts cannot be run in FreeBSD guest operating systems.



A default script for each power operation is included in VMware Tools. These scripts are located in the guest operating system in `/etc/vmware-tools`.

When You …	This Default Script Runs
Suspend the guest operating system	`suspend-vm-default`
Resume the guest operating system	`resume-vm-default`
Power off the guest operating system	`poweroff-vm-default`
Power on the guest operating system	`poweron-vm-default`

For each power state, you can use the default script or you can substitute a script you created. In addition, if you are logged on as root, you can edit a script, test a script or disable the running of a script. Complete the following steps.

1. To edit the appropriate power operation, deselect **Use default script to suspend guest operating system**, **Use default script to resume guest operating system**, **Use default script to shut down guest operating system** or **Use default script to power on guest operating system**.
2. Do one of the following:
   - To select a different script, click **Browse** and select the new script.

- To edit a script, click **Edit**. The script opens in `vi`. Make your changes there.

  Note: To edit scripts from the Scripts tab, `xterm` and `vi` must be installed in the guest operating system. The user trying to edit the script must be a root user and must have `vi` and `xterm` in his or her PATH. Otherwise, scripts can be edited manually in any text editor.

- To test a script, click **Test**.

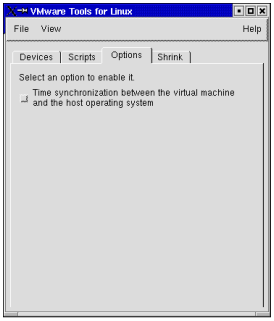
  Note: If you plan to test scripts in a Turbolinux 7.0 guest operating system, you need to update the Turbolinux guest operating system. This is a known issue with Turbolinux.

- To disable a script, select the path to the script and delete it.

3. Click **Apply** to save your settings.

Setting Options with VMware Tools

The Options tab gives you the option to synchronize the time in the guest operating system with the time in the host operating system.

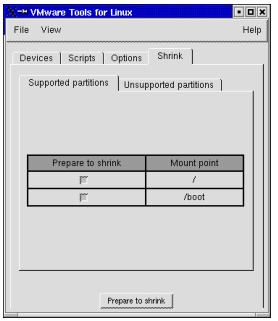


Note: You can synchronize the time in the guest operating system with the time in the host operating system only when the time in the guest is earlier than the time in the host.

To completely disable time synchronization, see Disabling Time Synchronization on page 91.

Shrinking Virtual Disks with VMware Tools

The Shrink tab gives you access to the controls you need if you wish to reclaim unused space in a virtual disk.



To shrink virtual disks, you should run VMware Tools as the root user (`su -`). This way, you ensure the whole virtual disk is shrunk. Otherwise, if you shrink the virtual disk as a non-root user, you cannot prepare to shrink the parts of the virtual disk that require root-level permissions.

For more information about shrinking virtual disks, see Defragmenting and Shrinking Virtual Disks on page 168.

Configuring VMware Tools in a NetWare Virtual Machine

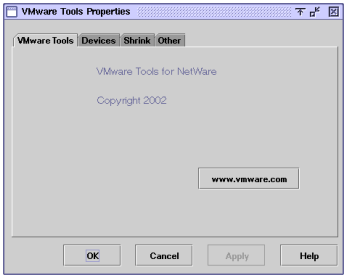
This section shows the options available in a NetWare 6.5, 6.0 or 5.1 guest. Since there is no graphical user interface for NetWare 4.2, there is no VMware Tools control panel as there is for newer NetWare guests. You can configure certain virtual machine options such as time synchronization, CPU idling and device configuration with VMware Tools in a NetWare 4.2 guest in the system console. For more information, see Configuring VMware Tools for NetWare Guests in the System Console on page 87.

Configuring VMware Tools in a NetWare 6.5, 6.0 or NetWare 5.1 Guest

To open the VMware Tools control panel, choose **Novell** > **Settings** > **VMware Tools for NetWare**.

Viewing Information About VMware Tools

For general information about VMware Tools, click the **VMware Tools** tab.

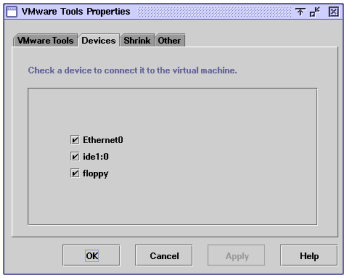


This tab contains:

- Copyright information.
- A button you click to visit the VMware Web site.

Connecting Devices with VMware Tools

The Devices tab allows you to enable or disable removable devices. Removable devices include the floppy and CD-ROM drives and the virtual network adapter.

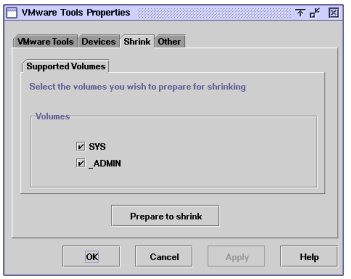


To connect a device, select the check box next to the device. To disconnect the device, deselect the check box next to the device.

Note: You can also set these options from the **VM** > **Removable Devices** menu in the virtual machine window.

Shrinking Virtual Disks with VMware Tools

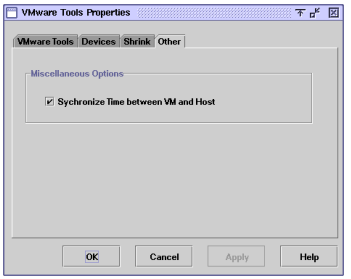
The Shrink tab gives you access to the controls you need if you wish to reclaim unused space in a virtual disk.



For more information about shrinking virtual disks, see Defragmenting and Shrinking Virtual Disks on page 168.

Setting Options with VMware Tools

The Other tab gives you the option to synchronize the time in the guest operating system with the time in the host operating system.



Note: You can synchronize the time in the guest operating system with the time in the host operating system only when the time in the guest is earlier than the time in the host.

To completely disable time synchronization, see Disabling Time Synchronization on page 91.

Configuring VMware Tools for NetWare Guests in the System Console

You can configure certain virtual machine options such as time synchronization, CPU idling and device configuration with VMware Tools in a NetWare virtual machine using the system console. The VMware Tools command line program is called `vmwtool`.

To see the options associated with this command, type `vmwtool help` at the system console.

When VMware Tools is installed in a NetWare guest, a heartbeat is always sent from the virtual machine to GSX Server. You can verify the virtual machine's heartbeat by viewing information about this virtual machine in the VMware Management Interface. For more information, see Monitoring the Virtual Machine's Heartbeat in the *VMware GSX Server Administration Guide*.

In addition, you can gracefully power the virtual machine on or off in the management interface. To power a virtual machine on or off with the management interface, see Changing a Virtual Machine's Power State from the Management Interface on page 120. Since there are no scripts for NetWare virtual machines, no scripts are run.

Summary of VMware Tools Commands for a NetWare Guest

Each command in the following table must be entered into the system console after the VMware Tools command `vmwtool`. Use the following format:

`vmwtool <command>`

`vmwtool` **Command**	**Definition**
`help`	Displays a summary of VMware Tools commands and options in a NetWare guest.
`partitonlist`	Displays a list of all disk partitions in the virtual disk and whether or not a partition can be shrunk.
`shrink <partition>`	Shrinks the listed partitions. If no partitions are specified, then all partitions in the virtual disk are shrunk. The status of the shrink process appears at the bottom of the system console. For more information, see Defragmenting and Shrinking Virtual Disks on page 168.
`devicelist`	Lists each removable device in the virtual machine, its device ID and whether the device is enabled or disabled. Removable devices include the virtual network adapter, CD-ROM and floppy drives.

`vmwtool` **Command**	**Definition**
`disabledevice <device name>`	Disables the specified device or devices in the virtual machine. If no device is specified, then all removable devices in the virtual machine are disabled. **Note:** You can also disable devices from the **VM** > **Removable Devices** menu in the virtual machine console window.
`enabledevice <device name>`	Enables the specified device or devices in the virtual machine. If no device is specified, then all removable devices in the virtual machine are enabled. **Note:** You can also enable devices from the **VM** > **Removable Devices** menu in the virtual machine console window.
`synctime [on|off]`	Lets you turn on or off time synchronization between the guest operating system and the host operating system. By default, time synchronization is turned off. Use this command without any options to view the current time synchronization status. You can synchronize the time in the guest operating system with time on the host operating system only when the time in the guest operating system is earlier than the time set in the host.
`idle [on|off]`	Lets you turn on or off the CPU idler. By default, the idler is turned on. The CPU idler program is included in VMware Tools for NetWare guests. The idler program is needed because NetWare servers do not idle the processor when the operating system is idle. As a result, a virtual machine takes processor time from the host regardless of whether the NetWare server software is idle or busy.

About the VMware Tools Service

When you install VMware Tools in a virtual machine, the VMware Tools service is one of the primary components installed. The service does the following:

- Synchronizes the time of the guest operating system with the time in the host operating system.
- Runs scripts in a virtual machine when the power state changes. See Executing Scripts When the Virtual Machine's Power State Changes on page 76.
- Executes commands in the virtual machine when you shut down or restart a Linux guest operating system.
- Sends a heartbeat to GSX Server so that it knows the guest operating system is running. A gauge for this heartbeat appears in the VMware Management Interface. For more information, see Using the Status Monitor in the *VMware GSX Server Administration Guide*.
- Passes messages from the host operating system to the guest operating system.
- Passes information between the guest operating system and a VMware Scripting API script.

The service starts automatically when you boot the guest operating system.

In a Windows guest, the VMware Tools service program file is called `VMwareService.exe`. Help is available by right-clicking the VMware Tools icon in the system tray and choosing **Help**.

In a Linux guest, the VMware Tools service is called `vmware-guestd`. To display help about the service, including a list of all options, use the following command:
`/etc/vmware/vmware-guestd --help`

This section covers the following topics.

- Synchronizing the Time in the Guest with the Host Operating System on page 91
- Executing Commands After You Power Off or Reset a Virtual Machine on page 92
- Passing a String from the Host Operating System to the Guest Operating System on page 92
- Passing Information Between the Guest Operating System and a VMware Scripting API Script on page 94

Synchronizing the Time in the Guest with the Host Operating System

The VMware Tools service can synchronize the date and time in the guest operating system with the time in the host operating system once every minute. To enable time synchronization for a Windows guest, see Setting Options with VMware Tools on page 78. To enable time synchronization for a Linux or FreeBSD guest, see Setting Options with VMware Tools on page 84. To enable time synchronization for a NetWare guest, see Setting Options with VMware Tools on page 87.

Synchronizing Guest Time in Response to System Events

The service synchronizes the date and time in the guest with the time in the host in response to various system events. These events include when you

- Take a snapshot. In the virtual machine's configuration file (`.vmx`), this setting is represented by the `time.synchronize.continue` option.
- Revert to a snapshot. In the virtual machine's configuration file (`.vmx`), this setting is represented by the `time.synchronize.restore` option.
- Resume a suspended virtual machine. In the virtual machine's configuration file (`.vmx`), this setting is represented by the `time.synchronize.resume.disk` option.
- Shrink the virtual disk. In the virtual machine's configuration file (`.vmx`), this setting is represented by the `time.synchronize.shrink` option.

Disabling Time Synchronization

If you want to completely disable time synchronization in the guest, open the virtual machine's configuration file (`.vmx`) in a text editor and set the following options to `FALSE`.

```
tools.syncTime
tools.synchronize.restore
time.synchronize.resume.disk
time.synchronize.continue
time.synchronize.shrink
```

Executing Commands After You Power Off or Reset a Virtual Machine

In a Linux guest, you can have the VMware Tools service execute specific commands when you shut down or restart the guest operating system. This is in addition to any script that you may have specified to run when you shut down the guest operating system.

In order to execute these commands, you need to modify `/etc/vmware-tools/tools.conf`. The commands are:

`halt-command = <command>`

(where `<command>` is the command to execute when you shut down the guest operating system)

`reboot-command = <command>`

(where `<command>` is the command to execute when you restart the guest operating system)

Passing a String from the Host Operating System to the Guest Operating System

With GSX Server and knowledge of a scripting language like Perl or NetShell (in a Windows 2000 guest operating system), you can pass a string from your virtual machine's configuration file in the host operating system to the guest operating system when you use the configuration file to launch a virtual machine.

What you pass to the guest operating system is up to you. You should pass a string only if you have a good understanding of a scripting language and know how to modify system startup scripts.

There are two ways of passing strings to a virtual machine's guest operating system:

1. You can place a string in the virtual machine's configuration file by setting the string to the `machine.id` parameter.

   For example, you can set this string:
   `machine.id = "Hello World."`
2. You pass the string to the guest operating system from the command line when you launch the virtual machine. See example 1 below.

You can pass items like the Windows system ID (SID), a machine name or an IP address. Inside the guest operating system startup script, you have the service retrieve this string, which can then be used in another script you write and include in the startup script to set your virtual machine's system ID, machine name or IP address.

This way, you can make copies of the same configuration file, add a different string to each (either in the configuration file itself or at the command line), then use these variations of the same configuration file to launch the same virtual disk in nonpersistent mode multiple times in a training or testing environment, for example.

This is what portions of two configuration files that point to the same virtual disk might look like. Each configuration file contains its own unique string set for the `machine.id` parameter.

`<config_file_1>.vmx` contains:

```
ide0:0.present = TRUE
ide0:0.fileName = "my_common_virtual_hard_drive.vmdk"
machine.id = "the_string_for_my_first_vm"
```

`<config_file_2>.vmx` contains:

```
ide0:0.present = TRUE
ide0:0.fileName = "my_common_virtual_hard_drive.vmdk"
machine.id = "the_string_for_my_second_vm"
```

Passing a string is also useful in situations where you want to deploy virtual machines on a network using a common configuration file, while providing each machine with its own unique identity. In this case, you specify the string at the command line (you need to launch each virtual machine with the `vmware -s` command) when you launch each virtual machine using this configuration file. See example 1 below.

Each virtual machine disk file must be copied into its own directory if it shares its filename with another virtual machine disk file.

The following example uses a Windows host and guest to illustrate how you can use the service to retrieve a string containing what will become the virtual machine's machine name and IP address. In this example, W2K-VM is the machine name and 148.30.16.24 is the IP address.

1. Define a string. Do this by either:
   - Adding the following line to your virtual machine's configuration file:
     `machine.id = "W2K-VM 148.30.16.24"`
     then launching a virtual machine using this configuration file.
   - Launching a virtual machine from the command line. At the command line, type:
     `"C:\Program Files\VMware\VMware GSX Server\vmware -s 'machine.id=W2K-VM 148.30.16.24' C:\Virtual Machines\win2000\win2000.vmx"`

Note: Write the above command on one line.

Note: On a Linux host, the machine ID passed on the command line takes precedence and is passed to the guest operating system if the following conditions are met:

- A virtual machine ID is specified in a configuration file.
- You use that file to launch a virtual machine.
- You also specify a machine ID on the command line.

2. Retrieve the string in the virtual machine. In a Windows guest, the command to retrieve the string is
   `VMwareService --cmd machine.id.get`

   Note: In your Linux guest operating system's startup script, add the following command before the network startup section:
   `/etc/vmware/vmware-guestd --cmd 'machine.id.get'`

You need to further customize this startup script so it uses the string the service retrieved during startup to set the virtual machine's network name to W2K-VM and its IP address to 148.30.16.24. This string should be located in the script before the network services are started. If you're using a Windows 2000 guest operating system, for example, you can call the NetShell utility (`netsh`) and pass it the contents of the string, which then uses the string accordingly (that is, it can set a new IP address for the virtual machine, if that is what was passed in the string originally).

From your host operating system, you can prevent a string from being passed to the guest operating system via the service. To do this, set the following line in your virtual machine's configuration file.

```
isolation.tools.getMachineID.disable = TRUE
```

Passing Information Between the Guest Operating System and a VMware Scripting API Script

When the guest operating system is running inside a virtual machine, the VMware Tools service allows you to pass information from a VMware Scripting API script you created (that is running in another host machine) to the guest operating system and from the guest operating system to a script.

For more information, go to the VMware Web site at *www.vmware.com/support/developer*.

CHAPTER 4

Running Virtual Machines

After you have installed VMware GSX Server, a guest operating system and VMware Tools, how do you run your virtual machine? The following sections give you highlights of the most common tasks.

- Overview of the VMware Virtual Machine Console Window on page 97
- Connecting to Virtual Machines and GSX Server Hosts on page 106
- Changing the Power State of a Virtual Machine on page 118
- Controlling the Virtual Machine Display on page 125
- Taking and Reverting to a Snapshot on page 129
- Running Virtual Machines from DVD-ROM or CD-ROM Discs on page 130
- Using PXE with Virtual Machines on page 132
- Sharing Files Between Guest and Host Operating Systems on page 134
- Installing Software in a Virtual Machine on page 139
- Cutting, Copying and Pasting Text on page 140
- Using Devices in a Virtual Machine on page 141

- Command Reference on page 145

For purposes of illustration, the examples in these sections use a Windows Server 2003 guest operating system. Some commands used in the illustrations are different from those used in other guest operating systems.

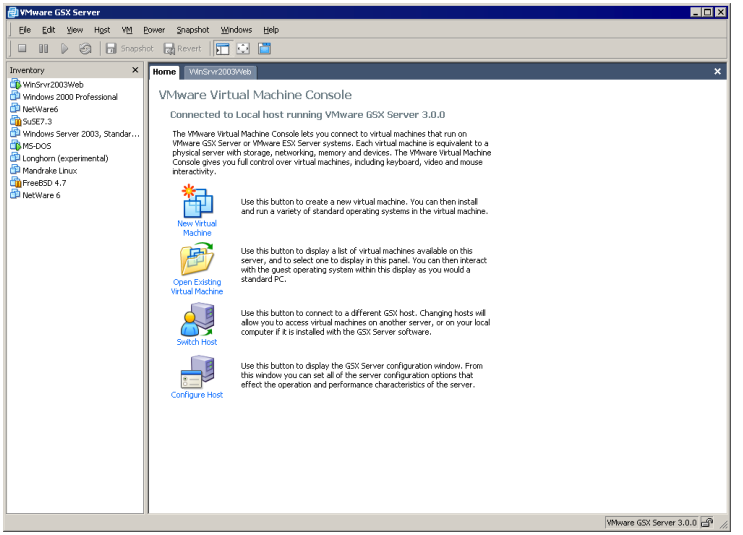
Overview of the VMware Virtual Machine Console Window

The following sections provide an overview of the VMware Virtual Machine Console:

- Using the Home Tab on page 98
- Using Tabs on page 102
- Configuring a Virtual Machine on page 103
- Using the Virtual Machine Inventory on page 103
- Displaying Hints on page 104
- Checking the Status of VMware Tools on page 104
- Creating a Screen Shot of a Virtual Machine on page 105

Think of a VMware GSX Server virtual machine as a separate computer that runs in a window on your physical computer's desktop. The VMware Virtual Machine Console lets you connect to multiple virtual machines and switch easily from one to another.

When you first connect a console to a GSX Server host, the Home tab appears in the virtual machine display. The Home tab indicates whether you are connecting to a GSX Server or ESX Server system, and the version of the server software. The status bar of the console window also displays this information.


VMware GSX Server
File Edit View Host VM Power Snapshot Windows Help
Snapshot Revert
Inventory
WinSrvr2003Web
Windows 2000 Professional
NetWare6
SuSE7.3
Windows Server 2003, Standar...
MS-DOS
Longhorn (experimental)
Mandrake Linux
FreeBSD 4.7
NetWare 6
Home
WinSrvr2003Web
VMware Virtual Machine Console
Connected to Local host running VMware GSX Server 3.0.0
The VMware Virtual Machine Console lets you connect to virtual machines that run on VMware GSX Server or VMware ESX Server systems. Each virtual machine is equivalent to a physical server with storage, networking, memory and devices. The VMware Virtual Machine Console gives you full control over virtual machines, including keyboard, video and mouse interactivity.
New Virtual Machine
Use this button to create a new virtual machine. You can then install and run a variety of standard operating systems in the virtual machine.
Open Existing Virtual Machine
Use this button to display a list of virtual machines available on this server, and to select one to display in this panel. You can then interact with the guest operating system within this display as you would a standard PC.
Switch Host
Use this button to connect to a different GSX host. Changing hosts will allow you to access virtual machines on another server, or on your local computer if it is installed with the GSX Server software.
Configure Host
Use this button to display the GSX Server configuration window. From this window you can set all of the server configuration options that effect the operation and performance characteristics of the server.
VMware GSX Server 3.0.0

If you are connecting to an older version of GSX Server or ESX Server, some of the controls and functionality of the interface change to accommodate the differences between the features available to that product. To see a list of what is different, see Connecting to Older GSX Server and ESX Server Systems and Older Virtual Machines on page 113.

New Menu Layouts

Menus in VMware GSX Server 3 are organized somewhat differently from those in VMware GSX Server 2. The following table lists the locations for the most commonly used menu items that have been moved:

Old Location	New Location
File > New > New Virtual Machine	File > New Virtual Machine
File > New > New Window	File > New Window
File > Detach and Exit	File > Exit
Devices > Removable Devices	VM > Removable Devices
Settings > Preferences (Windows) or Settings > Input Preferences (Linux)	Host > Settings (for global host settings) and Edit > Preferences (for user settings)
Settings > Configuration Editor	VM > Settings
Settings > Manage Virtual Networks	Host > Virtual Network Settings
Settings > VMware Tools Install	VM > Install VMware Tools
Settings > Upgrade Virtual Hardware	VM > Upgrade Virtual Hardware
Power > Send Ctrl+Alt+Del	VM > Send Ctrl+Alt+Del
Power > Grab Input	VM > Grab Input

Using the Home Tab

You can use the Home tab to quickly create new virtual machines, open existing virtual machines, connect to other GSX Server hosts and set global preferences for the current GSX Server host.

- For information on creating virtual machines, see Creating a New Virtual Machine with the New Virtual Machine Wizard on page 38.
- For information on opening an existing virtual machine, see Connecting to Virtual Machines and GSX Server Hosts on page 106.
- For information on changing hosts, see Connecting to a Different GSX Server Host on page 112.

- For information on configuring the GSX Server host, see Setting Global Preferences for VMware GSX Server in the *VMware GSX Server Administration Guide*.

Selecting virtual machines in the **Inventory** list opens them in new tabs. If the virtual machine is already running, its desktop appears in the virtual machine display.

WinSrvr2003Web - VMware GSX Server
File Edit View Host VM Power Snapshot Windows Help
Snapshot Revert
Inventory
WinSrvr2003Web
Windows 2000 Professional
NetWare6
SuSE7.3
Windows Server 2003, Standar...
MS-DOS
Longhorn (experimental)
Mandrake Linux
FreeBSD 4.7
NetWare 6
Home
WinSrvr2003Web
VMware is Virtual Infrastructure - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Address http://www.vmware.com
STORE DOWNLOAD SUPPORT COMPANY PARTNERS NEWS PRODUCTS SERVICES
vmware
AN EMC COMPANY
SEARCH
VMware is Virtual Infrastructure
NEWS
EMC Completes Acquisition of VMware
Industry's Leading SAN Vendors Demonstrate Broad VMware Support
VMware Unveils the Virtual Infrastructure, the First Step Towards Utility Computing
VMware Delivers VMware VirtualCenter
VMware Introduces the Industry's First Open Virtual Infrastructure Platform with the VMware SDK
VMware Receives Broad Industry
LEARN MORE
Open Letter to Customers
SERVER FOCUS: SEMINARS
Attend a free seminar for a discussion of the Virtual Infrastructure and an interactive demonstration of VMware VirtualCenter.
GARTNER ON VIRTUALIZATION
Find out why Gartner is saying 2004 will be the Year of
NOW AVAILABLE
VirtualCenter
LEARN MORE
VirtualCenter reduces the time, effort and complexity of data center management. Learn more today!
Virtual Infrastructure:
Learn how to transform your
WORKSTATION
Powerful Virtual Machine Software
BUY / TRY | UPGR
http://www.vmware.com/news/
Trusted sites
Start
VMware is Virtual Infr...
4:35 PM
VMware GSX Server 3.0.0

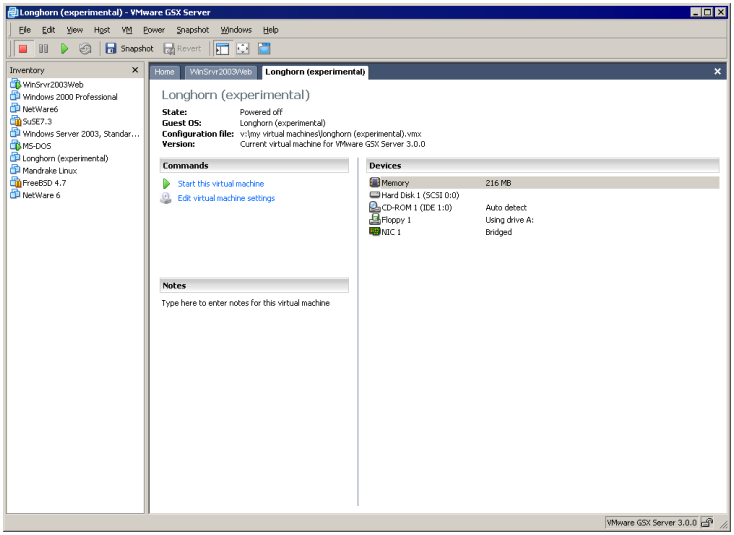
VMware Virtual Machine Console window on a Windows host
Right-click an icon for a removable device on the status bar to disconnect it or edit its configuration.

If the virtual machine is suspended or powered off, the virtual machine display lists information about the virtual machine, including its power state, the guest operating

One Window or Many — Your Choice

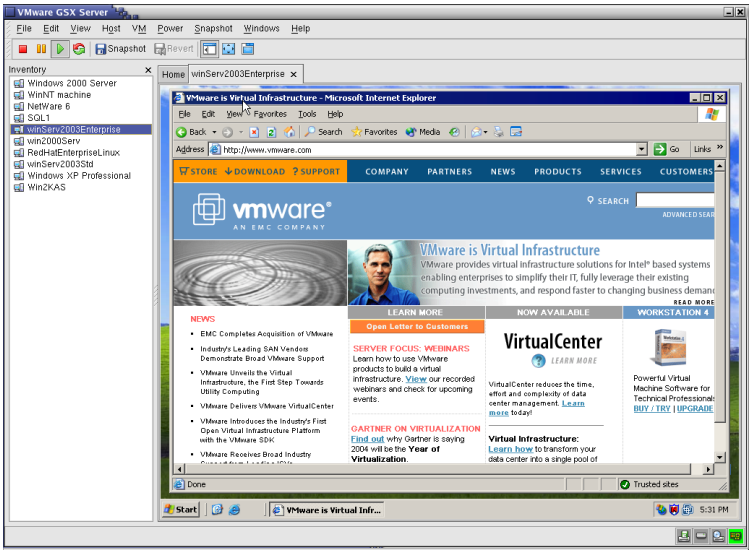
- In VMware GSX Server 3, you can open multiple virtual machines located on the same server host in the same console window. You can run multiple consoles and have each connect to virtual machines on different servers. Be sure you have enough memory and processor power to handle the number of virtual machines you want to run.

system, the location of the configuration file, and whether the virtual machine is configured for the current or an older version of GSX Server.



With the virtual machine powered off or suspended, you can enter notes about it, edit its settings or start it. Double-click on a device to configure it.

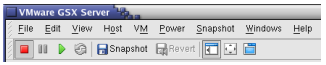
For information about the inventory, see Using the Virtual Machine Inventory on page 103.


VMware GSX Server
File Edit View Host VM Power Snapshot Windows Help
Snapshot Revert
Inventory
Windows 2000 Server
WinNT machine
NetWare 6
SQL1
winServ2003Enterprise
win2000Serv
RedHatEnterpriseLinux
winServ2003Std
Windows XP Professional
Win2KAS
Home
winServ2003Enterprise
VMware is Virtual Infrastructure - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Address http://www.vmware.com
STORE DOWNLOAD SUPPORT COMPANY PARTNERS NEWS PRODUCTS SERVICES CUSTOMERS
vmware
AN EMC COMPANY
SEARCH
VMware is Virtual Infrastructure
VMware provides virtual infrastructure solutions for Intel® based systems enabling enterprises to simplify their IT, fully leverage their existing computing investments, and respond faster to changing business demand
READ MORE
NEWS
EMC Completes Acquisition of VMware
Industry's Leading SAN Vendors Demonstrate Broad VMware Support
VMware Unveils the Virtual Infrastructure, the First Step Towards Utility Computing
VMware Delivers VMware VirtualCenter
VMware Introduces the Industry's First Open Virtual Infrastructure Platform with the VMware SDK
VMware Receives Broad Industry
LEARN MORE
Open Letter to Customers
SERVER FOCUS: WEBINARS
Learn how to use VMware products to build a virtual infrastructure. View our recorded webinars and check for upcoming events.
GARTNER ON VIRTUALIZATION
Find out why Gartner is saying 2004 will be the Year of Virtualization.
NOW AVAILABLE
VirtualCenter
LEARN MORE
VirtualCenter reduces the time, effort and complexity of data center management. Learn more today!
Virtual Infrastructure:
Learn how to transform your data center into a single pool of
WORKSTATION 4
Powerful Virtual Machine Software for Technical Professionals
BUY / TRY | UPGRADE
Done
Trusted sites
Start
VMware is Virtual Infr...
5:31 PM

VMware Virtual Machine Console window on a Linux host

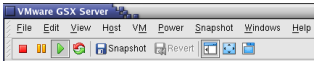
Instead of using physical buttons to turn this computer on and off, you use buttons on the toolbar at the top of the VMware Virtual Machine Console window.

Toolbar when a virtual machine is powered off (as seen on a Windows host)



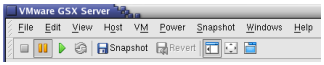
Toolbar when a virtual machine is powered off (as seen on a Linux host)

Toolbar when a virtual machine is powered on (as seen on a Windows host)



Toolbar when a virtual machine is powered on (as seen on a Linux host)

Toolbar when a virtual machine is suspended (as seen on a Windows host)



Toolbar when a virtual machine is suspended (as seen on a Linux host)

There are separate Power Off and Power On buttons. When you suspend a virtual machine, the Power On button becomes a Resume button.

Using Tabs

When a virtual machine is active, its virtual machine name appears on a tab at the top of the virtual machine display. To switch from one virtual machine display to another, click the tab of the virtual machine you want to see. It's like a soft KVM switch. You can use this feature in the windowed view and also in the quick switch view.

Tabs make it easy to switch among active virtual machines (as seen on a Windows host)

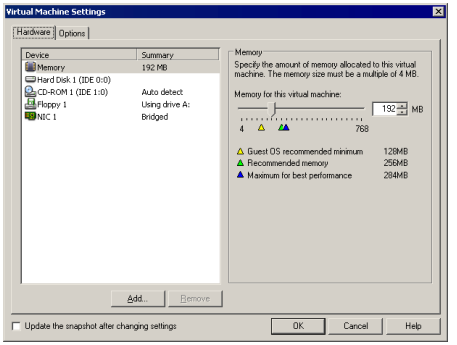
You can close a virtual machine's tab without interrupting the operation of the virtual machine. If the virtual machine is running when you close the tab, the virtual machine keeps running in the background and will be running when you open it in a tab again.

To view the virtual machine in the virtual machine display again, click the virtual machine in the inventory. If you close the Home tab, you can open it again by choosing **View** > **Go to Home Tab**.

If you want to view more than one virtual machine at the same time, you can open multiple console windows and launch one or more virtual machines in each. To view virtual machines on different servers, connect a new console to each server.

Configuring a Virtual Machine

To change settings for a virtual device, use the virtual machine settings editor. Choose **VM** > **Settings**, click the device name in the list on the left, then make changes on the right.



Use the virtual machine settings editor to add, remove and modify virtual machine components

The virtual machine settings editor (formerly the Configuration Editor) on Linux hosts now matches the virtual machine settings editor on Windows hosts.

For more information, see Using Devices in a Virtual Machine on page 141.

Using the Virtual Machine Inventory

When you create a virtual machine with GSX Server it is added to the inventory automatically. This allows the virtual machine to be accessed by a VMware Virtual Machine Console and the VMware Management Interface.

The inventory gives you a convenient way to open virtual machines. To add a virtual machine to the inventory (for example, if you copied the virtual machine from another host, you need to add it to the inventory manually), choose **File** > **Open Virtual Machine**, click **Browse** and browse to the virtual machine's configuration (`.vmx`) file.

Indicators on the icons for virtual machines in the list show whether a virtual machine is powered off, powered on or suspended.

To toggle the display of the inventory on or off, press `F9` or click the inventory button ( ) on the toolbar.

Removing a Virtual Machine from the Inventory

If you do not have a current need for a virtual machine, but do not want to delete it, you can remove it from the inventory instead. The virtual machine no longer appears in the console or the management interface.

Removing the virtual machine from the list does not affect the virtual machine's files. You can add the virtual machine to the list again at any time by using **File** > **Open Virtual Machine**.

To remove a name from the Inventory, take these steps.

1. Click a name in the list to select it.
2. Choose **VM** > **Remove from Inventory**.

Displaying Hints

GSX Server can display hints that appear in response to various actions you take when you run a virtual machine. The hints provide more information about these actions. By default, hints are turned off. If you want hints to appear, choose **Help** > **Hints** > **Show All Hints** in a console. VMware recommends that users new to GSX Server should display hints.

Each hint can be hidden on a case by case basis. When you want to hide a hint, check **Never show this hint again** before dismissing the hint dialog box.

Checking the Status of VMware Tools

For best performance, it is important to have VMware Tools installed and running in your virtual machine. For more information about VMware Tools, see Using VMware Tools on page 59.

After you install VMware Tools in a Windows virtual machine, the VMware Tools services start automatically when you start the guest operating system.

When VMware Tools is running in a Windows virtual machine, the VMware Tools icon appears in the system tray unless you disable the icon.

If the VMware Tools icon is not displayed in the system tray, you can use the VMware Tools control panel in the guest operating system (**Start** > **Settings** > **Control Panel** > **VMware Tools**) to change settings for VMware Tools. You can also reactivate the system tray icon. On the Options tab, check **Show VMware Tools in the taskbar**.

In a Linux or FreeBSD virtual machine, boot the guest operating system, start X and launch your graphical environment. Then you can launch the VMware Tools

background application with this command:

```
vmware-toolbox &
```

You may run VMware Tools as root or as a normal user. To shrink virtual disks, you must run VMware Tools as root (`su -`). To test and edit scripts, you must run VMware Tools as the root user.

In a NetWare 5.1 or higher guest operating system, you can access the VMware Tools control panel by choosing **Novell** > **Settings** > **VMware Tools for NetWare**.

With some window managers, you can place the command to start VMware Tools in a startup configuration so VMware Tools starts automatically when you start your graphical environment. Consult your window manager's documentation for details. For more information, see Starting VMware Tools Automatically on page 73.

A Reminder About Installing VMware Tools

An alert appears in the status bar — at the bottom left corner of the VMware GSX Server window — when your virtual machine is not running the version of VMware Tools that matches your version of GSX Server.

To launch the VMware Tools installer, choose **VM** > **Install VMware Tools**.

Note: Your guest operating system must be completely installed and running when you install VMware Tools.

For details, see Installing VMware Tools on page 62.

Creating a Screen Shot of a Virtual Machine

You can capture a screen shot of a virtual machine using **File** > **Capture Screen**. You can save this image as a bitmap (`.bmp`) file on a Windows host or as a portable network graphics (`.png`) file on a Linux host.

Connecting to Virtual Machines and GSX Server Hosts

This section covers the following topics:

- Connecting to a Virtual Machine from a Windows Host or Client on page 107
- Connecting to a Virtual Machine from a Linux Host or Client on page 110
- Connecting to a Virtual Machine from the VMware Management Interface on page 111
- Connecting to a Different GSX Server Host on page 112
- Connecting to Older GSX Server and ESX Server Systems and Older Virtual Machines on page 113

If you need to view a particular virtual machine's desktop, you can launch the VMware Virtual Machine Console and connect to the virtual machine.

Note: If you are connecting an older version of a console to the server, you need to select the virtual machine after you connect to the server host. VMware recommends you update the console on this client. You can download the appropriate installer on the Login or Status Monitor page of the VMware Management Interface. See Downloading the VMware Virtual Machine Console in the *VMware GSX Server Administration Guide*.

Netscape and Mozilla users must define a MIME type for the console first; Internet Explorer is automatically configured when the console is installed. For more information, see Setting a MIME Type to Launch the VMware Virtual Machine Console in the *VMware GSX Server Administration Guide*.

Connecting to a Virtual Machine from a Windows Host or Client

If you need to view a particular virtual machine's desktop, you can attach the VMware Virtual Machine Console and connect to the virtual machine.

1. Start the VMware Virtual Machine Console.

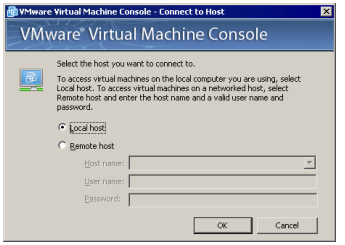
   Connecting to the Local GSX Server Host

   To quickly connect to the local host, double-click the VMware GSX Server Console icon on your desktop or choose **Start** > **Programs** > **VMware** > **VMware GSX Server**. Go to step 2.

   Connecting to any Server from a GSX Server Host

   To connect to the server of your choice from a GSX Server host, double-click the VMware Virtual Machine Console icon on your desktop or choose **Start** > **Programs** > **VMware** > **VMware Virtual Machine Console**.

   The VMware Virtual Machine Console - Connect to Host dialog box appears.

   
   VMware Virtual Machine Console - Connect to Host
VMware® Virtual Machine Console
Select the host you want to connect to.
To access virtual machines on the local computer you are using, select Local host. To access virtual machines on a networked host, select Remote host and enter the host name and a valid user name and password.
Local host
Remote host
Host name:
User name:
Password:
OK
Cancel

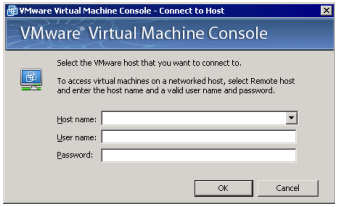
   Choose whether to connect to the local host or another GSX Server host.

   - To connect the console to a virtual machine on the local host, select **Local Host** then click **OK**.
   - To connect to a virtual machine on another GSX Server host, select **Remote Host**, specify the host name, user name and password to connect to that host, then click **OK**.

Connecting to the GSX Server Host from a Remote Client

If you are on a Windows client, double-click the VMware Virtual Machine Console icon on your desktop or choose **Start** > **Programs** > **VMware** > **VMware Virtual Machine Console**.

The VMware Virtual Machine Console - Connect to Host dialog box appears.

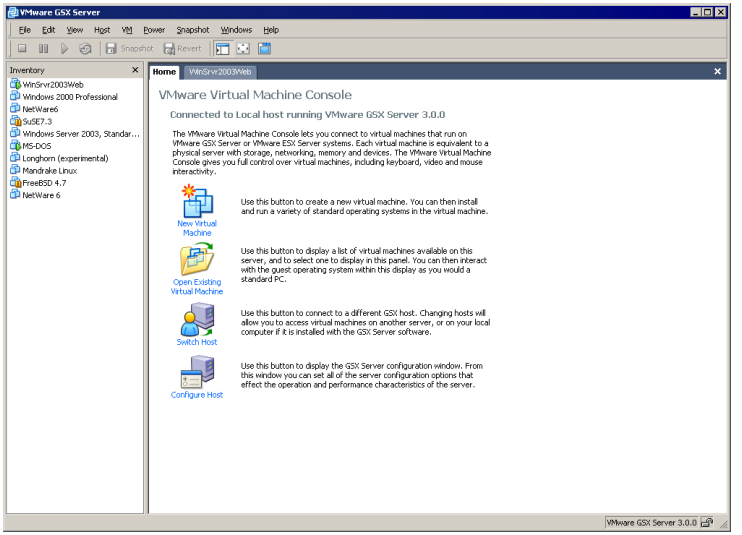

VMware Virtual Machine Console - Connect to Host
VMware® Virtual Machine Console
Select the VMware host that you want to connect to.
To access virtual machines on a networked host, select Remote host and enter the host name and a valid user name and password.
Host name:
User name:
Password:
OK
Cancel

Specify the host name, user name and password to connect to that host, then click **OK**.

2. If this is the first time you have launched GSX Server and you did not enter the serial number when you installed the product (an option available on a Windows host), you are prompted to enter it. The serial number is on the registration card in your package or in the email message that came with your electronic distribution. Enter your serial number and click **OK**.

   The serial number you enter is saved and GSX Server does not ask you for it again. For your convenience, GSX Server automatically sends the serial number to the VMware Web site when you use certain Web links built into the product (for example, **Help** > **VMware on the Web** > **Register Now!** and **Help** > **VMware on the Web** > **Request Support**). This allows us to direct you to the correct Web page to register and get support for your product.

The VMware Virtual Machine Console window opens.


VMware GSX Server
File Edit View Host VM Power Snapshot Windows Help
Inventory
WinSrvr2003Web
Windows 2000 Professional
NetWare6
SuSE7.3
Windows Server 2003, Standar...
MS-DOS
Longhorn (experimental)
Mandrake Linux
FreeBSD 4.7
NetWare 6
Home
WinSrvr2003Web
VMware Virtual Machine Console
Connected to Local host running VMware GSX Server 3.0.0
The VMware Virtual Machine Console lets you connect to virtual machines that run on VMware GSX Server or VMware ESX Server systems. Each virtual machine is equivalent to a physical server with storage, networking, memory and devices. The VMware Virtual Machine Console gives you full control over virtual machines, including keyboard, video and mouse interactivity.
New Virtual Machine
Use this button to create a new virtual machine. You can then install and run a variety of standard operating systems in the virtual machine.
Open Existing Virtual Machine
Use this button to display a list of virtual machines available on this server, and to select one to display in this panel. You can then interact with the guest operating system within this display as you would a standard PC.
Switch Host
Use this button to connect to a different GSX host. Changing hosts will allow you to access virtual machines on another server, or on your local computer if it is installed with the GSX Server software.
Configure Host
Use this button to display the GSX Server configuration window. From this window you can set all of the server configuration options that effect the operation and performance characteristics of the server.
VMware GSX Server 3.0.0

3. Select the name of the virtual machine you want to use in the **Inventory** list at the left of the console window.

   If the virtual machine you want to use does not appear in the inventory, choose **File** > **Open Virtual Machine**, click **Browse** and browse to the configuration (`.vmx`) file for the virtual machine you want to use.

   Note: By default, GSX Server stores virtual machines in `<installdrive>:\Virtual Machines\<guestOS>`.

4. If the virtual machine is not running, click the **Power On** button to start it.

5. If VMware Tools is not running in the virtual machine, click anywhere inside the virtual machine window to give the virtual machine control of your mouse and keyboard.

6. If you need to log on, type your name and password just as you do on a physical computer except that instead of using Ctrl-Alt-Del to log on, use Ctrl-Alt-Ins. If you use Ctrl-Alt-Del, the Windows host detects the command.

Connecting to a Virtual Machine from a Linux Host or Client

If you need to view a particular virtual machine's desktop, you can attach the VMware Virtual Machine Console and connect to the virtual machine.

You need an X server to run the VMware Virtual Machine Console. If an X server is not installed, you must install `libxpm.so.4`, located on your Linux distribution disk.

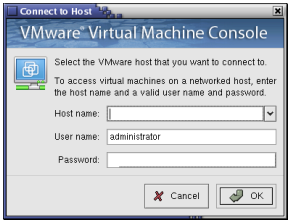
1. Start the VMware Virtual Machine Console. Open a terminal window, and do one of the following:

   - To connect a console to a virtual machine on the local host, type
     `vmware &`

     Then press Enter.

   - To connect a console from a client to a virtual machine on a remote host, type
     `vmware-console &`

     Then press Enter. The Connect to Host dialog box appears.

     
     Connect to Host
VMware Virtual Machine Console
Select the VMware host that you want to connect to.
To access virtual machines on a networked host, enter the host name and a valid user name and password.
Host name:
User name:
administrator
Password:
Cancel
OK

     Specify the host name, user name and password to connect to that host, then click **OK**.

2. If this is the first time you have launched the console, a dialog box asks if you want to rename existing virtual disks using the new `.vmdk` extension. Click **OK** to search all local drives on the host computer and make this change.

   The converter also renames the files that store the state of a suspended virtual machine, if it finds them. It changes the old `.std` file extension to `.vmss`. However, it is best to resume and shut down all suspended virtual machines before you upgrade GSX Server.

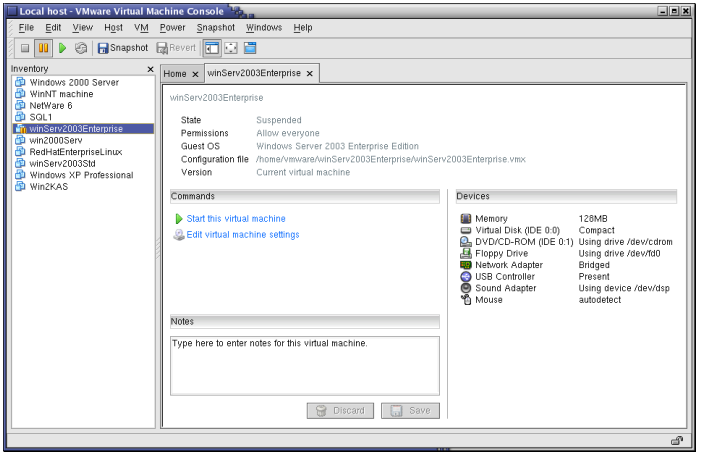
   Besides renaming files, the converter updates the corresponding virtual machine configuration files so they identify the virtual disks using the new filenames.

   If you plan to store your virtual disk files or suspended state files on a Windows Server 2003 host in the future, it is important to convert the filenames to avoid conflicts with the System Restore feature of Windows Server 2003.

Linux Hosts: One Chance to Rename Disk Files

- The Rename Virtual Disks dialog box appears only once. If you click **Cancel**, you do not have another opportunity to update the filenames and configuration files automatically.

The VMware Virtual Machine Console window opens.



3. Select the name of the virtual machine you want to use in the **Inventory** list at the left of the console window.

   If the virtual machine you want to use is not shown in the Inventory, choose **File** > **Open Virtual Machine**, click **Browse** and browse to the configuration file (`.vmx` or `.cfg` file) for the virtual machine you want to use.

   Note: By default, GSX Server stores virtual machines in `/var/lib/vmware/Virtual Machines/<guestOS>`.

4. If the virtual machine is not running, click the **Power On** button to start it.
5. If VMware Tools is not running in the virtual machine, click anywhere inside the virtual machine display to give the virtual machine control of your mouse and keyboard.
6. If you need to log on, type in your name and password just as you do on a physical computer.

Connecting to a Virtual Machine from the VMware Management Interface

If you need to view a particular virtual machine's desktop, you can attach the VMware Virtual Machine Console and connect to the virtual machine.

From the management interface, click the terminal icon ( ) in the row for the virtual machine to which you want to connect with the console. For more information on connecting the console, see Connecting to a Virtual Machine from a Windows Host or

Client on page 107 and Connecting to a Virtual Machine from a Linux Host or Client on page 110.

Connecting to a Different GSX Server Host

Each VMware Virtual Machine Console can connect to one GSX Server host at a time. If you need to connect to a virtual machine on another host, you can launch another console and connect to any virtual machine on the new host. Or you can switch hosts in the same console window. To switch hosts from a console, complete the following steps.

1. From a VMware Virtual Machine Console, choose **Host** > **Switch Host**. The Switch Host dialog box appears.
2. Choose whether to connect to the local host or another GSX Server host.
   - To connect the console to a virtual machine on the local host, select **Local Host** then click **OK**.

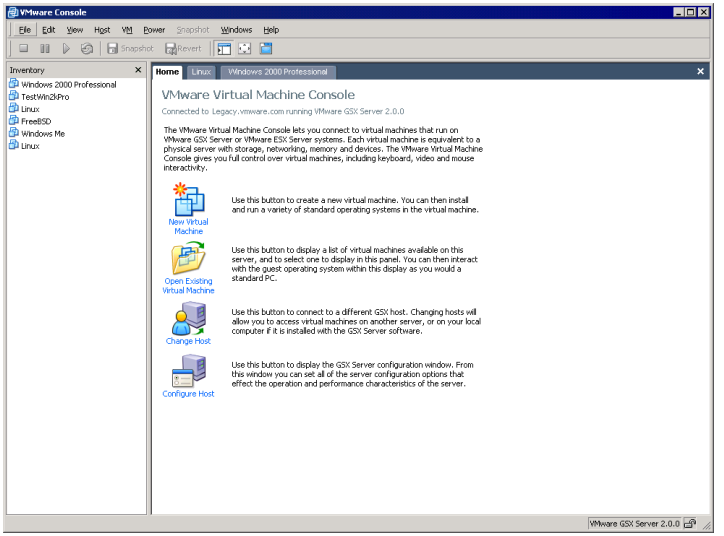
     Note: If you are connecting to the host from a client, you cannot choose between the local or a remote host, as you can connect only to a remote host.
   - To connect to a virtual machine on another GSX Server host, select **Remote Host**, specify the host name, user name and password to connect to that host, then click **OK**.

If you were already connected to a different GSX Server host, you lose that connection.

Connecting to Older GSX Server and ESX Server Systems and Older Virtual Machines

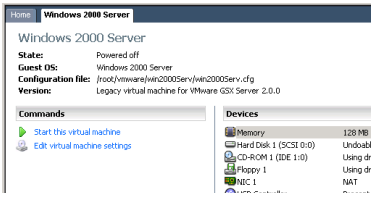
When you connect a VMware Virtual Machine Console from GSX Server 3 to an older virtual machine or older GSX Server or ESX Server system, the console controls adapt to the older machine or server. Certain capabilities, such as creating new virtual machines or configuring the host, are unavailable when you connect to an older virtual machine or server. Features introduced in GSX Server 3 — such as snapshots — are not available to virtual machines on older servers.

As discussed in Overview of the VMware Virtual Machine Console Window on page 97, the Home tab and the status bar in the console window display the type (whether the server is a GSX Server or ESX Server system) and version of the server to which you are connecting.


VMware Console
File Edit View Host VM Power Snapshot Windows Help
Snapshot Revert
Inventory
Windows 2000 Professional
TestWin2kPro
Linux
FreeBSD
Windows Me
Linux
Home
Linux
Windows 2000 Professional
VMware Virtual Machine Console
Connected to Legacy.vmware.com running VMware GSX Server 2.0.0
The VMware Virtual Machine Console lets you connect to virtual machines that run on VMware GSX Server or VMware ESX Server systems. Each virtual machine is equivalent to a physical server with storage, networking, memory and devices. The VMware Virtual Machine Console gives you full control over virtual machines, including keyboard, video and mouse interactivity.
New Virtual Machine
Use this button to create a new virtual machine. You can then install and run a variety of standard operating systems in the virtual machine.
Open Existing Virtual Machine
Use this button to display a list of virtual machines available on this server, and to select one to display in this panel. You can then interact with the guest operating system within this display as you would a standard PC.
Change Host
Use this button to connect to a different GSX host. Changing hosts will allow you to access virtual machines on another server, or on your local computer if it is installed with the GSX Server software.
Configure Host
Use this button to display the GSX Server configuration window. From this window you can set all of the server configuration options that effect the operation and performance characteristics of the server.
VMware GSX Server 2.0.0

If you are connected to a virtual machine created under an earlier version of GSX Server, ESX Server or Workstation, the virtual machine's summary information indicates that it is a **legacy** virtual machine; if the virtual machine was created under GSX Server 3, ESX Server 2 or Workstation 4, the virtual machine is considered to be a

current virtual machine when run under GSX Server 3. Look for the **Version** information in the virtual machine display when the virtual machine is not running.



In addition, the virtual machine settings editor identifies a virtual machine as a legacy virtual machine.

Note: To determine which version of another VMware product is older than GSX Server 3, see the *VMware Virtual Machine Mobility Planning Guide* on the VMware Web site. This guide also discusses moving virtual machines between VMware products.

If you are running an older virtual machine on a GSX Server 3 host, the virtual machine is considered a legacy virtual machine until you upgrade the virtual hardware. Some legacy virtual machine settings are disabled. For example, you cannot add physical disks to a legacy virtual machine on a current GSX Server host.

The abilities and limitations of connecting a console to older servers and older virtual machines are outlined below.

Configuring the Host

You cannot configure an older GSX Server host or any ESX Server system.

Creating and Deleting Virtual Machines

You cannot create new virtual machines on the older host from the console. You cannot delete virtual machines from the older host from a console.

Browsing for Virtual Machines

You cannot browse for virtual machines on an older GSX Server host. You can connect only to those virtual machines that appear in the Open Virtual Machine dialog box.

Connecting to an ESX Server System

When you connect the console to virtual machines on an ESX Server system, the virtual machine settings editor allows you to change settings for removable devices only. All other settings are read-only.

You cannot configure an ESX Server system.

Upgrading Virtual Hardware

Using the console, you can upgrade the virtual hardware of a legacy virtual machine to the virtual hardware level supported by GSX Server 3.

You cannot upgrade the virtual hardware for a legacy virtual machine to the level of hardware used by the older version of GSX Server on which the virtual machine is running. For example, you cannot upgrade the hardware of a virtual machine created under GSX Server 1 to the hardware supported by GSX Server 2.

Once you upgrade the virtual hardware, the virtual machine is no longer considered to be a legacy virtual machine.

Virtual Disk Modes

You can specify a disk mode for the virtual disk — persistent, undoable or nonpersistent. For a discussion of disk modes, see *www.vmware.com/support/gsx25/doc/disks_modes_gsx.html*.

You cannot choose independent mode, as you can with GSX Server 3 virtual machines.

If you are running a legacy virtual machine on a GSX Server 3 host, the virtual machine's disk modes are honored, but cannot be modified. Independent disk modes do not apply (see Independent Disks on page 164). For information on how snapshots work with disk modes, see Snapshots and Legacy Disk Modes on page 158.

Adding New Virtual Disks

If you add a new virtual disk, you must always split the disk into 2GB files.

Using Snapshots

You cannot take snapshots of virtual machines running on older hosts.

You can take snapshots of legacy virtual machines running on a GSX Server 3 host. You cannot update the snapshot of a legacy virtual machine, nor can you specify what to do with the snapshot when you power off the virtual machine. For more information, see Snapshots and Legacy Disk Modes on page 158.

Virtual CD-ROM Drive Differences

You can enable raw access for the virtual machine. This is known as legacy emulation in GSX Server 3. You cannot use the DVD-ROM or CD-ROM drive on the client.

You cannot connect the DVD-ROM or CD-ROM drive exclusively to a legacy virtual machine that runs on a GSX Server 3 host. Using legacy emulation is the same as if the virtual machine were running on a GSX Server 2 host and you disabled raw access. If you are connecting to the virtual machine from a client, you cannot use the client's DVD-ROM or CD-ROM drive.

Virtual Network Interface Card (NIC)
If you are connected to a GSX Server 2 virtual machine, you can choose the `vmxnet` adapter only if the guest operating system is Windows 2000, Windows XP or Windows Server 2003.

Furthermore, if you are connected to a GSX Server 2.0 or 2.0.1 virtual machine on a GSX Server 2.5 or higher host and you choose the `vmxnet` adapter, you are prompted to install VMware Tools. This installs the GSX Server 2.5 version of VMware Tools, which supports the `vmxnet` adapter.

If you are connected to a GSX Server 2.0 or 2.0.1 host, you cannot choose the `vmxnet` adapter at all.

Virtual Network Settings
If you are connected to a legacy virtual machine on a GSX Server 3 host from a remote client, you cannot configure its virtual network settings.

Virtual Parallel Port
You can enable bidirectional mode for virtual parallel ports for a virtual machine running on an older host. For more information, see *www.vmware.com/support/gsx25/doc/devices_parallel_gsx.html*.

If the virtual machine is running on a GSX Server 3 host, you cannot enable bidirectional mode, as this is no longer required by the new virtual parallel port implementation.

General Virtual Machine Options
You can enable repeatable resume for the virtual machine. To enable or disable repeatable resume, choose **VM** > **Settings** > **Options** > **General** and check or clear the **Enable repeatable resume** check box. For more information, see *www.vmware.com/support/gsx25/doc/running_repres_gsx.html*.

If you are running a legacy virtual machine on a current host, you cannot change the guest operating system selection.

Virtual Machine Permissions
If the virtual machine is on an older host, you cannot set permissions for it to be accessible to all users.

If a legacy virtual machine is running on a current host, you can change this setting.

Advanced Virtual Machine Settings
You cannot specify process priorities for the virtual machine. For more information, see Adjusting Priorities for Virtual Machine Processes (Windows Hosts Only) in the *VMware GSX Server Administration Guide*.

If the virtual machine uses physical (raw) disks and is on an older server, you can hide read-only partitions so the virtual machine can see only partitions to which it can write data. You cannot do this if the virtual machine is running on a GSX Server 3 host.

Virtual Sound Adapter
You can add a virtual sound adapter to a virtual machine on an older host, but you cannot configure it.

If a legacy virtual machine is running on a GSX Server 3 host, it can use only the default host sound adapter. The sound adapter does not work when you connect to this virtual machine from a remote client.

Virtual Machine Display
You can configure the display depth for a virtual machine on an older host.

You cannot configure the display depth for a legacy virtual machine on a current host.

Product Messages
Messages that the console displays are based on the version of the server to which you are connecting. References to menu items, interface elements and product terminology are relevant to that server type and version, not necessarily to the current version of GSX Server.

Entering the Serial Number
You cannot enter the serial number for an older host.

Viewing the Tip of the Day
You cannot view the tip of the day when you are connected to an older host.

Changing the Power State of a Virtual Machine

The following topics discuss ways you change a virtual machine's power state:

- Using Power Options for Virtual Machines on page 118
- Suspending and Resuming Virtual Machines on page 121
- Shutting Down a Virtual Machine on page 121
- Powering Virtual Machines On and Off When the Host Starts and Shuts Down on page 122

Using Power Options for Virtual Machines

The basic power operations for a virtual machine include powering on, powering off, suspending, resuming and resetting. These options are analogous to the power operations on a physical computer.

If VMware Tools is running, you can run scripts when you change the power state of a virtual machine. For more information, see Executing Scripts When the Virtual Machine's Power State Changes on page 76.

When you reset a virtual machine, you can choose either to restart the guest operating system, which gracefully closes applications and restarts the guest operating system, or to reset the virtual machine, which is the same as pressing the reset button on a physical computer.

Similarly, when you power off the virtual machine, you can choose either to shut down the guest operating system, which gracefully closes applications and shuts the guest operating system down, or to turn off the virtual machine, which is the same as pressing the power button on a physical computer.

All the power options are available on the **Power** menu. The menu items may not be available, depending upon the current power state of the virtual machine. For example, if the virtual machine is powered off, you cannot select any power off, suspend, resume or reset options.

For the purpose of running scripts in the guest operating system, the commands on the **Power** menu take precedence over how the toolbar power buttons are configured.

For example, if the Suspend toolbar button is configured to run a script when you suspend the virtual machine, and you do not want to run the script, choose **Power** > **Suspend**. Similarly, if the Suspend toolbar button is not configured to run a script, and

you want to run the script at the time you suspend the virtual machine, choose **Power** > **Suspend after running script**.

Options for Powering On a Virtual Machine

You can choose from the following options when powering on a virtual machine:

- **Power On** — powers on the virtual machine. This is the same as clicking the **Power On** button on the toolbar. When the virtual machine is suspended, this menu item appears as **Resume**.
- **Power On and run script** — powers on the virtual machine, then executes the associated script.

Options for Powering Off a Virtual Machine

You can choose from the following options when powering off a virtual machine:

- **Power Off** — powers off the virtual machine. This is similar to turning off a physical computer by pressing its power button, so any programs running in the virtual machine may be adversely affected.
- **Shut Down Guest** — runs the associated script, then gracefully shuts the guest operating system down and, if the guest operating system supports Advanced Power Management, powers off the virtual machine. This is the same as choosing **Start** > **Shut Down** > **Shut Down** in a Windows operating system or issuing a `shutdown` command in a Linux operating system.

The stop button ( ) on the toolbar can be configured to power off the virtual machine or shut down the guest operating system. Choose **VM** > **Settings**, then click **Options** > **Power**, and choose the desired action in the list under **Power Controls**.

Options for Suspending a Virtual Machine

You can choose from the following options when suspending a virtual machine:

- **Suspend** — suspends the virtual machine.
- **Suspend after running script** — executes the associated script, then suspends the virtual machine.

Options for Resuming a Virtual Machine

You can choose from the following options when resuming a virtual machine:

- **Resume** — resumes the suspended virtual machine. When the virtual machine is powered off, this menu item appears as **Power On**.
- **Resume and run script** — resumes the suspended virtual machine, then executes the associated script.

Options for Resetting a Virtual Machine

You can choose from the following options when resetting a virtual machine:

- **Reset** — resets the virtual machine. This is similar to resetting a physical computer by pressing its reset button, so any programs running in the virtual machine may be adversely affected.
- **Restart Guest** — gracefully restarts the virtual machine. This is the same as choosing **Start** > **Shut Down** > **Restart** in a Windows operating system or issuing a `reboot` command in a Linux operating system.

The reset button ( ) on the toolbar can be configured to reset the virtual machine or restart the guest operating system. Choose **VM** > **Settings**, then click **Options** > **Power**, and choose the desired action in the list under **Power Controls**.

Changing a Virtual Machine's Power State from the Management Interface

Depending upon your permissions, you can change the power state of the virtual machine in the management interface. Your permissions are listed in the **Users and Events** tab for the virtual machine. For more information, see Viewing a List of Connected Users in the *VMware GSX Server Administration Guide*.

To change the virtual machine's power state, click the button that corresponds to the virtual machine's current power state. A popup menu appears, displaying the following buttons:

Button	Description
	Shuts down the guest operating system and powers off the virtual machine. Any open applications close, the guest operating system shuts down, then GSX Server powers off the virtual machine. VMware Tools executes any script associated with this power state change. When this icon is red, the virtual machine is powered off.
	Suspends a running virtual machine. VMware Tools executes any script associated with this power state change. When this icon is amber, the virtual machine is suspended.
	Powers on a stopped virtual machine or resumes a suspended virtual machine. VMware Tools executes any script associated with this power state change. When this icon is green, the virtual machine is running.
	Restarts a guest operating system. Any open applications close, the guest operating system shuts down, then GSX Server restarts the guest. VMware Tools executes any script associated with this power state change.

Changing the power state executes any script associated with the power state change. For more information about running scripts, see Executing Scripts When the Virtual Machine's Power State Changes on page 76.

Suspending and Resuming Virtual Machines

You can save the current state of your virtual machine by suspending it. Later, you can resume the virtual machine to pick up work quickly, right where you stopped — with all documents you were working on open and all applications in the state they were at the time you suspended the virtual machine.

To suspend a virtual machine:

1. If your virtual machine is running in full screen mode, return to window mode by pressing the Ctrl-Alt key combination.
2. Click **Suspend** on the console toolbar.

To resume a virtual machine that you have suspended:

1. Start the VMware Virtual Machine Console and choose a virtual machine you have suspended. The process is the same as that described in Connecting to Virtual Machines and GSX Server Hosts on page 106.
2. Click **Resume** on the console toolbar.

   Note that any applications you were running at the time you suspended the virtual machine are running and the content is the same as it was when you suspended the virtual machine.

For more information, see Suspending and Resuming Virtual Machines on page 150.

To suspend and resume a virtual machine from the VMware Management Interface, see Changing a Virtual Machine's Power State from the Management Interface on page 120.

Shutting Down a Virtual Machine

As with physical computers, you need to shut down your guest operating system before you power off your virtual machine. Follow the standard steps you also follow in the host operating system.

For example, in a Windows guest operating system, take these steps.

1. Select **Shut Down** from the **Start** menu of the guest operating system (inside the virtual machine).
2. Select **Shut Down**, then click **OK**.
3. After the guest operating system shuts down, you can turn off the virtual machine. Click **Power Off**.

Powering Virtual Machines On and Off When the Host Starts and Shuts Down

You can configure a virtual machine to power on automatically when the GSX Server host starts. When the host shuts down, you can specify whether you want to power off the virtual machine or shut down the guest operating system.

You can enable these settings as long as the startup and shutdown options are not disabled for the server. See Configuring Startup and Shutdown Options for Virtual Machines in the *VMware GSX Server Administration Guide*.

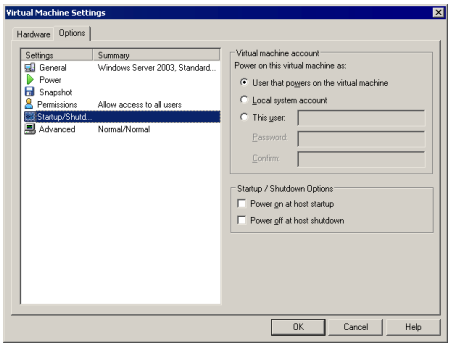
In order for you to specify these options, the virtual machine must be configured to run as either the local system account or as a specific user. The virtual machine cannot be configured to run as the user that powers it on.

You must power off the virtual machine to change the startup and shutdown options.

Setting Startup and Shutdown Options from the Console

To configure a virtual machine's startup and shutdown options from the console, complete the following steps.

1. In the VMware Virtual Machine Console, select the virtual machine, then choose **VM** > **Settings**. The virtual machine settings editor opens.
2. Click the **Options** tab, then click **Startup/Shutdown**.



3. Under **Startup/Shutdown Options**, choose whether you want this virtual machine to power on automatically when the GSX Server host starts up, and whether you want to power off the virtual machine or shut down the guest operating system when the host shuts down.

   To power on the virtual machine when the host starts, select **Power on the virtual machine** in the **On host startup** list.

To power off the virtual machine — or shut down the guest — when the host shuts down, select the appropriate option in the **On host shutdown** list.

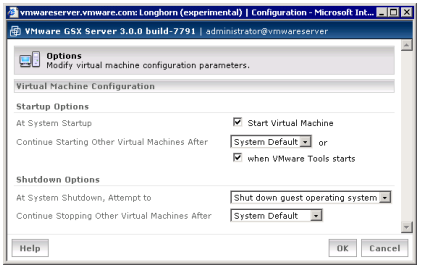
4. Click **OK** to save your changes and close the virtual machine settings editor.

You cannot configure a virtual machine to start up or shut down automatically when the host starts or shuts down until the host is configured accordingly. To configure the host, see Configuring Startup and Shutdown Options for Virtual Machines in the *VMware GSX Server Administration Guide*. You must log on to the management interface as an administrator to configure the GSX Server host.

Setting Startup and Shutdown Options from the Management Interface

To configure a virtual machine's startup and shutdown options from the management interface, complete the following steps.

1. On the Status Monitor page of the management interface, click the virtual machine menu icon ( ), then choose **Configure Options**. The Options tab for the virtual machine appears.
2. Click **Edit** under **System Startup Options** or **System Shutdown Options**. The Options page appears.



3. To allow the virtual machine to start up when the system starts up, check the **Start Virtual Machine** check box.

   Specify the period of time before the next virtual machine starts in the **Continue Starting Virtual Machines After** list. Choose the number of minutes or whether GSX Server should not wait before starting the next virtual machine. If you select **Other**, a prompt appears; specify in the prompt the number of minutes to wait. The **System Default** is specified in Configuring Startup and Shutdown Options for Virtual Machines in the *VMware GSX Server Administration Guide*.

   To specify that VMware Tools should start in a virtual machine before the next virtual machine starts, check the **when VMware Tools starts** check box. If

VMware Tools does not start in the virtual machine before the specified time elapses, GSX Server starts the next virtual machine.

4. Specify what happens to the virtual machine when the system shuts down. In the **At System Shutdown, Attempt to** list, select whether you want to power off the virtual machine, shut down the guest operating system or suspend the virtual machine.

   Configure when GSX Server should stop the next virtual machine after this virtual machine stops in the **Continue Stopping Other Virtual Machines After** list. Choose the number of minutes, if any, that GSX Server should wait before stopping the next virtual machine. If you select **Other**, a prompt appears; specify in the prompt the number of minutes to wait. The **System Default** is specified in Configuring Startup and Shutdown Options for Virtual Machines in the *VMware GSX Server Administration Guide*.

5. Click **OK** to save your settings.
6. Click **Close Window** to return to the virtual machine's Options page.

Controlling the Virtual Machine Display

There are a variety of ways for you to control how your virtual machines display in a console window. They include:

- Using Full Screen Mode on page 125
- Using Quick Switch Mode on page 125
- Taking Advantage of Multiple Monitors on page 126
- Fitting the Console Window to the Virtual Machine on page 127
- Fitting a Windows Guest Operating System's Display to the VMware Virtual Machine Console Window on page 127
- Simplifying the Screen Display on page 128

Using Full Screen Mode

Virtual machines run faster in full screen mode.

If you want your virtual machine's display to fill the screen — so you no longer see the borders of the VMware Virtual Machine Console window — click the **Full Screen** button on the toolbar. You can also use a keyboard shortcut — press the Ctrl-Alt-Enter keys at the same time.

To get out of full screen mode — to show your virtual machine inside a VMware Virtual Machine Console window again — press the Ctrl-Alt key combination.

You can configure the virtual machine to enter full screen mode when you power it on. In a console, choose **VM** > **Settings**, then click **Options** > **Power**. Select the **Enter full screen mode after powering on** check box, then click **OK** to save your setting.

Note: GSX Server does not support running virtual machines in full screen mode on dual-monitor systems.

Using Quick Switch Mode

Quick switch mode is similar to full screen mode, except it adds tabs at the top of the screen for switching from one active virtual machine to another. The virtual machine's display resizes to fill the screen completely, except for the space occupied by the tabs.

To enter quick switch mode, choose **View** > **Quick Switch**.

To view the VMware GSX Server menus and toolbar while you are using quick switch mode, move the mouse pointer to the top of the screen.

To resize a Windows guest operating system's display so it fills as much of the screen as possible in quick switch mode, choose **View** > **Fit Guest to Window**. The Fit Guest

to Window option works only if you have the current version of VMware Tools installed in the guest operating system and you disabled Autofit.

Note: When you choose **Fit Guest to Window**, VMware GSX Server adjusts the display settings of your Windows guest operating system as needed. If you subsequently run the virtual machine in window mode, you may want to change the display settings back to their previous values.

To get out of quick switch mode, move the mouse pointer to the top of the screen to activate the menu, then choose **View** > **Quick Switch**.

Taking Advantage of Multiple Monitors

If your host has a standard multiple monitor display, you can run separate sets of virtual machines on each of the monitors. To use two monitors, launch two instances of the VMware Virtual Machine Console. Start one or more virtual machines in each console window, then drag each console window to the monitor on which you want to use it. For the largest possible screen display, switch each of the windows to quick switch mode (**View** > **Quick Switch**).

To switch mouse and keyboard input from the virtual machine on the first screen to the virtual machine on the second screen, move the mouse pointer from one to the other. You do not need to take any special steps if VMware Tools is running in both guest operating systems and if you are using the default settings for grabbing input. If you have changed the defaults, you may need to press Ctrl-Alt to release the mouse pointer from the first virtual machine, move it to the second virtual machine, then click in the second virtual machine so it grabs control of mouse and keyboard input.

Note: Multiple monitor support is experimental in this release of VMware GSX Server. It does not work properly with some third-party desktop management software or display drivers.

Note: If you switch to full screen mode, VMware GSX Server always uses the primary display. To use multiple monitors, you must use either the normal (windowed) mode or quick switch mode.

Fitting the Console Window to the Virtual Machine

The **View** menu gives you two ways to adjust the size of the VMware Virtual Machine Console window so it exactly fits the virtual machine's display.

Autofit is toggled on or off each time you click it. When **Autofit** is on, the console window adjusts automatically to fit the virtual machine's display. When it is off, you can adjust the console window to a size of your choice. If you make the console window smaller than the virtual machine's display, scroll bars appear so you can move to the part of the virtual machine's display that you want to see.

If **Autofit** is off, you can choose **View** > **Fit** to adjust the console window so it fits the virtual machine's display.

Fitting a Windows Guest Operating System's Display to the VMware Virtual Machine Console Window

If your Windows guest operating system is set to a display resolution larger or smaller than the size of the virtual machine window, you can make it fit exactly by choosing **View** > **Fit Guest to Window**.

When you choose **Fit Guest to Window**, GSX Server adjusts the display settings of your Windows guest operating system as needed. If you subsequently run the virtual machine in window mode, you may want to change the display settings back to their previous values.

Note: When you use the Fit Guest to Window option and the window is small, your guest operating system's screen resolution may be set to something smaller than VGA (640 x 480). Some installers and other programs do not run at resolutions smaller than 640 x 480. If either the width or height is smaller than the corresponding dimension required for VGA, the programs refuse to run. Error messages may include such phrases as "VGA Required To Install" or "You must have VGA to install."

There are two ways to work around this problem.

- If your host computer's screen resolution is high enough, you can enlarge the window, then choose **Fit Guest to Window**.
- If your host computer's screen resolution does not allow you to enlarge the window enough, do not use **Fit Guest to Window**. Instead, set the guest operating system's screen resolution to 640 x 480 or larger.

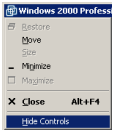
Simplifying the Screen Display

If you prefer, you can hide many of the controls visible in the console window.

Use the **View** menu to toggle the following controls on or off:

- Inventory
- Toolbar
- Status bar
- Virtual machine tabs

On a Windows host, you can also hide the menu bar. To do so, click the title bar icon, then choose **Hide Controls**.



Choosing **Hide Controls** hides the menu bar, the toolbar, the status bar and the inventory.

For the simplest possible VMware Virtual Machine Console window on a Windows host, first choose **View** > **Virtual Machine Tabs** to turn off the tabs. Then, from the title bar icon shortcut menu, choose **Hide Controls**.

Windows 2000 Professional - VMware GSX Server
VMware: Enterprise-Class Virtualization Software - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Address http://www.vmware.com
STORE DOWNLOAD SUPPORT
COMPANY PARTNERS NEWS PRODUCTS CUSTOMERS
vmware
SEARCH
ADVANCED SEARCH
Enterprise-class virtual machine software
More than 5,000 customers and 80% of the Fortune 100 use VMware software to flexibly scale and control their computing infrastructure to quickly respond to business demands, improve return on investment and dramatically reduce total cost of ownership.
SERVER FOCUS: WEBINARS
Learn how to use VMware server products to reduce your TCO. Enroll in upcoming events, or view past events in the archive.
VMWARE SOLUTIONS FOR:
Software Development & Testing
Server Consolidation
Disaster Recovery
Training
Sales Demos
More...
NEWS
Stopping Intel Sprawl - VMware Lets Users Consolidate Proliferating Boxes
VMware Announces ESX Server 2 For The Most Demanding Production Environments
VMware ESX SERVER 2 LETS ENTERPRISES:
• Partition and Consolidate servers in High-Performance environments
• Scale and Control Computing Resources
• Deploy SMP Apps in Virtual Machines
WORKSTATION 4
Powerful Virtual Machine Software for Technical Professionals
http://www.vmware.com/download/
Internet
Start
VMware: Enterprise-Cl...
4:01 PM

Using the View menu and the title bar icon shortcut menu, you can remove all visible controls from the VMware Virtual Machine Console window. Only the virtual machine display is in view.

Taking and Reverting to a Snapshot

GSX Server lets you take a snapshot of a virtual machine at any time and revert to that snapshot at any time.

You can take a snapshot while a virtual machine is powered on, powered off or suspended. A snapshot preserves the virtual machine just as it was when you took the snapshot — the state of the data on all the virtual machine's disks and whether the virtual machine was powered on, powered off or suspended.

When you revert to a snapshot, you discard all changes made to the virtual machine since you took the snapshot.

Use the **Snapshot** and **Revert** buttons on the console toolbar to take a snapshot and revert to it later.

You can take a new snapshot at any time. When you do so, you replace the previous snapshot. You can have only one active snapshot at a time.

For more information, including examples of ways you can use the snapshot, see Taking Snapshots on page 153.

Running Virtual Machines from DVD-ROM or CD-ROM Discs

You can store a virtual disk on DVD-ROM or CD-ROM, and run the virtual machine from your GSX Server host's DVD/CD-ROM drive. You do not have to copy the virtual disk files from the DVD-ROM or CD-ROM to the GSX Server host.

One suggested use for this method is to install GSX Server on a host you want to use for product demonstrations, which could be a laptop. Instead of taking up limited hard disk space with virtual disks, you can have any number of virtual machines with virtual disks burned onto DVD-ROM or CD-ROM and point each virtual machine's configuration file to the virtual disk on the DVD-ROM or CD-ROM.

Other uses include sales or proof-of-concept demonstrations where you want to keep virtual disk files off a customer's system but want to illustrate a multiple machine demonstration in the customer's environment. Or you can have multiple physical servers in a datacenter run virtual machines without copying the virtual disk files to the servers themselves. Yet another use is, if you need a "master" virtual machine for some purpose, you can create a write-protected copy of your original virtual machine.

The virtual disk must be an independent disk in nonpersistent mode, since any changes you make in the virtual machine cannot be written to the DVD-ROM or CD-ROM. The redo log for the virtual machine must be on the GSX Server host. For more information about independent disks, see Independent Disks on page 164.

Similarly, if you want to take a snapshot of the virtual machine, the redo log for the virtual machine must be on the GSX Server host before you take the snapshot. For more information about snapshots, see Taking Snapshots on page 153.

Note: If you take a snapshot of the virtual machine and you want to save the changes made to the virtual disk after the snapshot was taken, you must copy the virtual disk to the GSX Server host's hard drive, then update the snapshot. Otherwise, you can keep appending changes to the redo log. In addition, if you copy the disk file to a Windows host, you need to make the disk file writable.

Before you run a virtual machine with a virtual disk stored on DVD-ROM or CD-ROM, you should consider whether you may need to modify the virtual machine's BIOS at some point. In this case, the virtual machine's BIOS, which is stored in a file called `nvram`, must be located on the GSX Server host. Or, you can add a setting to the virtual machine's configuration file that allows for the `nvram` file to be on the DVD-ROM or CD-ROM, where it cannot be modified.

Note: The performance of the virtual machine accessing a virtual disk stored on a DVD-ROM or CD-ROM depends upon the speed of the DVD-ROM/CD-ROM drive. Keep in mind that a virtual machine on a DVD-ROM/CD-ROM drive runs slower than it would if it were running on your host's hard disk.

To run a virtual machine with a virtual disk stored on DVD-ROM or CD-ROM, complete the following steps.

1. Create a virtual machine and install the guest operating system and any applications you need within it.
2. Make sure the virtual machine is powered off. Burn the virtual disk (`.vmdk`) files onto a DVD-ROM or CD-ROM. Place the DVD-ROM or CD-ROM into the GSX Server host's DVD-ROM or CD-ROM drive.
3. Choose **VM** > **Settings** to open the virtual machine settings editor for this virtual machine. On the **Hardware** tab, select **Virtual Disk** and browse to the virtual disk file on the DVD-ROM or CD-ROM.
4. Click **Advanced**. Under **Mode**, check **Independent** and set the disk mode to **Nonpersistent**. Click **OK** to save these settings.
5. On the Options tab, select **General**. Under **Working directory**, browse to and select a location for the redo log on the GSX Server host.
6. Click **OK** to save your changes. The virtual machine settings editor closes.
7. In a text editor, open the virtual machine's configuration file (`.vmx`) and add two of the following lines to the file:

   `disk.locking = FALSE`

   `nvram = <path on GSX Server host>\nvram` (if you think you need to modify the virtual machine's BIOS)
   or
   `nvram.mode = "nonpersistent"` (if you do not need to modify the virtual machine's BIOS)
8. Save your changes and close the configuration file.

The virtual machine is now ready to be run with the virtual disk on the GSX Server host's DVD-ROM or CD-ROM drive.

Note: Another method you can use is to burn all virtual machine files (the configuration file, `nvram` and virtual disk files) onto DVD-ROM or CD-ROM. First make sure the redo log directory points to a drive on your GSX Server host and that the configuration file has all the desired settings before you burn the files onto the DVD-ROM or CD-ROM.

Using PXE with Virtual Machines

You can use a preboot execution environment (commonly known as PXE) to boot a virtual machine over a network. When you use PXE with a virtual machine, you can:

- Remotely install a guest operating system over a network without needing the operating system installation media.
- Deploy an image of a virtual disk to the virtual machine.
- Boot a Linux virtual machine over the network and run it diskless.

You use PXE with your virtual machine in conjunction with remote installation tools such as Windows 2000 Remote Installation Services or the Red Hat Linux 9.0 installer's PXE package. You can use Ghost or Altiris to stream an image of an already configured virtual disk to a new virtual machine.

Make sure the virtual machine has a virtual network adapter; one is installed by default. VMware supports PXE when the virtual machine is configured to use either the `vmxnet` or `vlance` virtual network adapter.

The virtual machine must have a virtual disk without a guest operating system installed.

When a virtual machine boots and there is no guest operating system installed, it proceeds to boot from devices (hard disk, CD-ROM drive, floppy drive and network adapter) in the order in which they occur in the boot sequence specified in the virtual machine's BIOS. If you plan to use PXE with a virtual machine, it is a good idea to put the network adapter at the top of the boot order. When the virtual machine first boots, press F2 to enter the virtual machine's BIOS and change the boot order there.

As the virtual machine boots from the network adapter, it tries to connect to a DHCP server. The DHCP server provides the virtual machine with an IP address and a list of any PXE servers available on the network. After the virtual machine connects to a PXE server, it can connect to a bootable disk image (such as an operating system image or a Ghost or Altiris disk image) and start installing a guest operating system.

VMware has tested and supports the following PXE configurations with GSX Server 3:

- Remote installation of a Windows Server 2003 guest operating system from a server running Windows Server 2003 Automated Deployment Services
- Remote installation of a Windows 2000 guest operating system from a server running Windows 2000 Server/Advanced Server Remote Installation Services
- Remote installation of a Linux guest operating system from a Red Hat Enterprise Linux 3.0 AS PXE boot server

- Remote installation of a supported guest operating system from a Ghost image using Windows 2000 and Ghost RIS Boot package
- Remote installation of a supported guest operating system from an Altiris image using a Windows 2000 Altiris server
- Network booting a Linux virtual machine by connecting with the Linux Diskless option to a Red Hat Enterprise Linux 3.0 AS server

Sharing Files Between Guest and Host Operating Systems

To share files between a host computer and a virtual machine or between two virtual machines, you use the networking features of GSX Server. If you know how to share files between two physical computers on a network, you already know how to share files with a virtual machine.

This section describes four scenarios for sharing files between two systems, either a host computer and a virtual machine or two virtual machines, where

- Both systems run Windows operating systems, using Windows file sharing
- You are connecting from a Linux system to a Windows system, using `smbmount`
- You are connecting from a Windows system to a Linux system, using Samba
- Both systems run Linux operating systems, using NFS, FTP and Telnet

You can apply the same principles to share files between virtual machines.

The following scenarios assume you have set up your virtual machine using NAT networking. Besides giving the virtual machine a direct connection to the host computer's network, NAT networking sets up a virtual network adapter on the host computer. You can use this adapter, which connects to a virtual switch identified as `vmnet8`, to communicate between host and virtual machine. You can also connect two or more virtual machines using `vmnet8`. For details on NAT networking, see Network Address Translation (NAT) on page 232.

In all cases, the user name you use to log on to the system from which you are connecting must be a user on the system to which you want to connect.

To understand how to share files, read the following sections:

- Sharing Files Between Two Windows Systems on page 135
- Sharing Files by Connecting to a Windows System from a Linux System on page 135
- Sharing Files by Connecting to a Linux System from a Windows System on page 136
- Sharing Files Between Two Linux Systems on page 138

Sharing Files Between Two Windows Systems

To share files between two Windows systems (where one machine is a host and the other is a virtual machine, or both are virtual machines), be sure the file and printer sharing service is installed for both operating systems and the folders you want to share are marked as shared. Then you can browse from one system to the shared folder or folders on the other system.

Sharing Files by Connecting to a Windows System from a Linux System

To share files on a Windows system with a Linux system (by connecting to a Windows host from a Linux guest or connecting to a Windows guest from a Linux host or guest), you can mark a folder as shared on the Windows system, then use the `smbmount` utility in the Linux system to mount the shared folder. For example, if you want to share the folder `C:\docs` on a Windows 2000 system called `win2k` with a Linux system at `/mnt/docs`, follow the steps below. You may want to set up a shell script to run these commands.

1. Set up the folder or folders to share on the Windows system.
2. Create a user account on the Windows system for the Linux system user name that you are using to connect to the Windows system.

   Otherwise, if you know the user name and password for a user account that can access the Windows system, you can specify that account on the command line.
3. From your Linux system, log on as root.

   ```
   su -
   ```
4. Add the Windows system's host name and IP address to the `hosts` file, if the system cannot be found by name.
5. Mount the Windows share on your Linux system. Enter the following command all on one line.

   ```
   mount -t smbfs -o username=<Windows system user
   account>,password=<password> //win2k/docs /mnt/docs
   ```

   (Substitute the appropriate host name, share and mount point for your systems.)

Note: If you do not want to expose this password on the command line or in a script, leave out that option and provide the password when prompted after you run the command.

Now you are connected to the shared folder on the Windows system from your Linux system and can begin to share files between the two.

Sharing Files by Connecting to a Linux System from a Windows System

To share files on a Linux system with a Windows system (by connecting to a Linux host from a Windows guest or connecting to a Linux guest from a Windows host or guest), you can run Samba on the Linux system and browse shared directories in the Linux file system from Network Neighborhood in the Windows system.

You need to modify Samba on the Linux host operating system so it recognizes the `vmnet8` switch, otherwise you cannot access the Linux file system. You need to do this even if you installed host-only networking (as Samba is installed when you install host-only networking with GSX Server). For more information about Samba, see Using Samba for File Sharing on a Linux Host on page 280.

Connecting to a Linux Host from a Windows Guest

If you want to share the directory `/home/user/shared`, for example, on a Linux host operating system with a Windows guest operating system, follow these steps:

1. On the Linux host operating system, back up the `smb.conf` file to a file called something like `smb.conf.orig`.
   ```
   cd /etc/vmware/vmnet1/smb
   cp smb.conf smb.conf.orig
   ```
2. Modify Samba on the Linux host system. Edit the following lines in `/etc/vmware/vmnet1/smb/smb.conf`.
   a. Comment out the line starting with `interfaces=<IP addresses>`.
   b. Below this line, add `interfaces=vmnet1 vmnet8`.
   c. Provide a network workgroup name. Set `workgroup=<name>`.
   d. If you do not want to use the standard DNS name for the Linux system, set `netbiosname=<Linux system name>`.
   e. You can leave `security=user`, unless you cannot connect, in which case use `security=share`.
   f. Set `encrypt passwords=yes`.
   g. In the `[global]` section, define a different shared memory access key. Add this line:
   ```
   sysv shm key=/dev/vmnet8
   ```
   h. For better performance, find this line:
   ```
   socket options = TCP_NODELAY
   ```
   Edit the line to state:

```
socket options = TCP_NODELAY SO_RCVBUF=8192
SO_SNDBUF=8192
```

Note: This setting must be entered on one line.

i. To create the share, add the following:
```
[SHARE_NAME]
path = /home/user/shared
public = no
writable = yes
printable = no
```
(since you want to share files, not a printer)

j. Save this file and create a backup copy to protect these changes when you upgrade GSX Server.

3. Restart the Samba services to load the new settings.

   If GSX Server is running on the Linux host system, suspend or shut down all running virtual machines and close all GSX Server console windows.

   On the Linux host operating system, at a command prompt, type

   ```
   /etc/init.d/vmware restart
   ```

   On some Linux distributions, the command is

   ```
   /etc/rc.d/init.d/vmware restart
   ```

   Connect to the virtual machine with a console and run the Windows guest operating system from which you want to connect to the Linux host. The user ID you use to log on to the Windows guest must be in the Linux host's `smbpasswd` file. If you use the same user name and password to log on to the guest as you do on the Linux host, then you are not prompted to log on when you browse the Linux host.

   If you are connecting to the Linux system from a Windows Me, Windows 98 or Windows 95 guest operating system, NetBEUI must be installed in the guest operating system before you can browse the file system. If you need to install NetBEUI, you may need your Windows installation CD-ROM.

   When the system restarts, the Samba service does not appear in the list of services starting up, but it does start, unless an error appears.

Connecting to a Linux Guest from a Windows Host or Guest

To share the directory `/home/user/shared`, for example, on a Linux guest operating system with a Windows host or guest operating system, follow these steps:

1. On the Linux guest operating system, back up the `smb.conf` file to a file called something like `smb.conf.orig`.
   ```
   cp /etc/smb.conf /etc/smb.conf.orig
   ```
2. Modify Samba on the Linux system to share the directory. To create the share, add the following to `/etc/smb.conf`.
   ```
   [SHARE_NAME]
   path = /home/user/shared
   public = no
   writable = yes
   printable = no
   ```
   (since you want to share files, not a printer)
3. Restart the Samba services to load the new settings. On the Linux guest operating system, at a command prompt, type:
   ```
   /etc/init.d/smb restart
   ```
   On some Linux distributions, the command is
   ```
   /etc/rc.d/init.d/smb restart
   ```
   When the system restarts, the Samba service appears in the list of services starting up.

Sharing Files Between Two Linux Systems

To share files between two Linux systems (where one machine is a host and the other is a virtual machine, or both are virtual machines), you can use NFS on the system to connect to and the `nfsmount` utility in the system from which you are making the connection.

As with any Linux network, you can use NFS, FTP or Telnet to connect from one Linux system (either virtual or physical) to another Linux system (either virtual or physical).

Installing Software in a Virtual Machine

Installing software in a virtual machine is just like installing it on a physical computer. For example, to install software in a Windows virtual machine, take the following steps:

1. Be sure you have started the virtual machine and, if necessary, logged on. In the console window, check **VM** > **Removable Devices** to be sure the virtual machine has access to the CD-ROM drive and, if needed, the floppy drive.
2. Insert the installation CD-ROM or floppy disk into the proper drive on the GSX Server host. If you are installing from a CD-ROM, the installation program may start automatically.
3. If the installation program does not start automatically, click the Windows **Start** button, go to **Settings** > **Control Panel**, then double-click **Add/Remove Programs** and click the **Install** button. Follow the instructions on screen and in the user manual for your new software.

Note: Some applications use a product activation feature that creates a key, based on the virtual hardware in the virtual machine where it is installed. Changes in the configuration of the virtual machine may require you to reactivate the software. To minimize the number of significant changes, set the final memory size for your virtual machine and install VMware Tools before you activate the software.

Note: When you try to run a few programs — including the installer for the Japanese-language version of Trend Micro Virus Buster — GSX Server may appear to hang. To work around this problem, try disabling acceleration in the guest. For more information, see Issues Installing or Running Applications in a Guest Operating System in the *VMware GSX Server Administration Guide*.

Cutting, Copying and Pasting Text

When VMware Tools is running, you can cut (or copy) and paste text between applications in the virtual machine and the host computer or between two virtual machines. Use the normal hot keys or menu choices to cut, copy and paste.

Note: If you are copying text from a Windows host into a Linux guest operating system, you can paste only by using the middle mouse button. If you are using a two-button mouse, click both mouse buttons at the same time to paste.

To turn off this feature — to prevent accidental copying and pasting from one environment to another — change your preferences.

Choose **Edit** > **Preferences**. On the Input tab, clear the **Enable copy and paste to and from virtual machine** check box.

Using Devices in a Virtual Machine

The following sections provide an overview on the devices in your virtual machine.

- Adding, Configuring and Removing Devices in a Virtual Machine on page 141
- Connecting and Disconnecting Removable Devices on page 144

Adding, Configuring and Removing Devices in a Virtual Machine

The virtual machine settings editor (**VM** > **Settings**) is the control center where you can add devices to a virtual machine, change the settings for those devices and remove them. In addition, you can add, change and remove devices in the VMware Management Interface.

For more information about adding and configuring devices such as parallel ports, serial ports, USB controllers and generic SCSI devices, see Configuring Devices on page 289.

For information on adding and configuring virtual disks, physical disks, DVD-ROM and CD-ROM drives and floppy drives, see Using Disks in a Virtual Machine on page 161.

For information on adding and configuring virtual network adapters, see Adding and Modifying Virtual Network Adapters on page 238.

For information on configuring virtual machine memory, see Allocating Memory to a Virtual Machine on page 340.

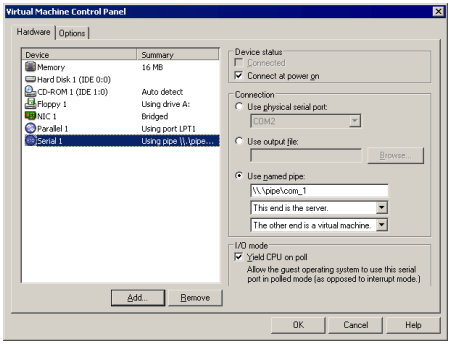
To remove a device or other hardware from a virtual machine, make sure it is powered off. You can remove hardware using the console or the management interface.

Note: You cannot add or remove some items from a virtual machine, such as the processor, SCSI controllers or the virtual display adapter. GSX Server creates SCSI controllers as needed when you add SCSI devices. However, the number of virtual SCSI controllers is included in the six virtual PCI slot limit for a virtual machine. For information on which devices use PCI slots, see Virtual Machine Specifications on page 24.

Removing Hardware Using the Console

To remove hardware from a virtual machine, make sure the virtual machine is powered off, then complete the following steps.

1. In a console, select the virtual machine, then click **Edit virtual machine settings**. The virtual machine settings editor appears.

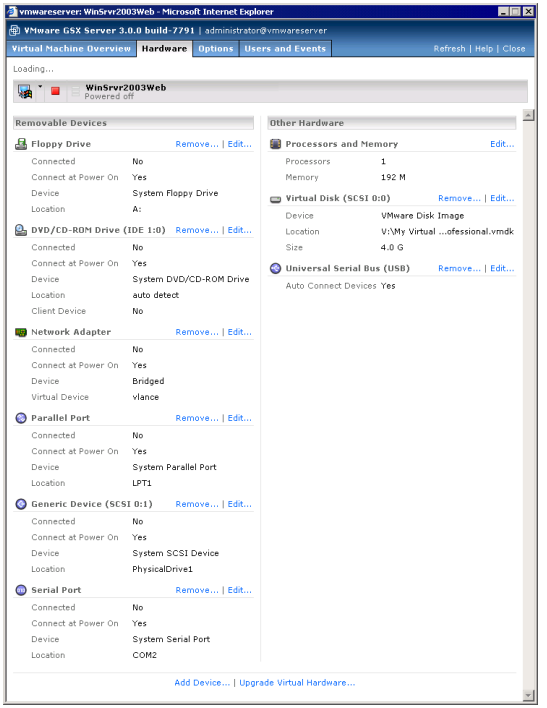


2. Select the item you want to remove, then click **Remove**.
3. Click **OK** to save the change and close the virtual machine settings editor.

Removing Hardware Using the Management Interface

To remove hardware from a virtual machine, make sure the virtual machine is powered off, then complete the following steps.

1. In the management interface, on the Status Monitor page, click the virtual machine menu icon ( ) for the virtual machine, then choose **Configure Hardware**. The Hardware page appears.



2. Next to the item you want to remove, click **Remove**. You are asked for confirmation before the device is removed.

Connecting and Disconnecting Removable Devices

Choose **VM** > **Removable Devices** to connect and disconnect removable devices that you have configured for a virtual machine — including floppy drives, DVD/CD-ROM drives, USB devices and Ethernet adapters — while the virtual machine is running.

When you choose **VM** > **Removable Devices**, a submenu appears. Choose a device from that menu to connect or disconnect it, and to edit device settings. If you choose **Edit**, a dialog box appears. Make all the changes you want to make, then click **OK**.

Command Reference

The following sections describe command line options that are available when you launch a console and keyboard shortcuts that you can use while a virtual machine is running.

- Startup Options on a Linux Host on page 145
- Startup Options on a Windows Host on page 146
- Using Keyboard Shortcuts on page 147

Startup Options on a Linux Host

The following list describes various options available when you run GSX Server from the command line on a Linux host operating system. You can also set the power options when you change a virtual machine's power options. See Using Power Options for Virtual Machines on page 118.

```
vmware [-x] [-X] [-q] [-v] [-s] [-l]
[/<path_to_config>/<config>.vmx]
[X toolkit options]
```

`-x` automatically powers on the virtual machine when a console is launched. This is equivalent to clicking the **Power On** button in the console toolbar.

`-X` automatically powers on the virtual machine, then switches the console window to full screen mode.

Note: This option does not work when you connect with the console from a remote client to the GSX Server host.

`-q` closes the virtual machine's tab when the virtual machine powers off. If no other virtual machine is open, it also closes the console. This is particularly useful when the guest operating system is capable of powering off the virtual machine.

`-l` launches the console, connecting it directly to the local host.

`-v` displays the product name, version and build number.

`-s NAME=VALUE` sets a configuration variable called `name` to `value`. This configuration setting applies until the virtual machine is powered off. These settings are found in the virtual machine's configuration (`.vmx`) file. You should use this option only if you know the exact variable and value to use; typically you use this when you are troubleshooting issues, and VMware support suggests you use a particular configuration setting.

`-m` automatically starts the virtual machine in quick switch mode. This option works for virtual machines running on Linux hosts only. For information on quick switch mode, see Using Quick Switch Mode on page 125.

`/<path_to_config>/<config>.vmx` (or `.cfg`) launches a virtual machine using the specified configuration file.

X toolkit options can be passed as arguments, although some of them (most notably the size and title of the console window) cannot be overridden.

Options to Use when Connecting Remotely

If you are connecting to a virtual machine from a remote client, you can use the following options:

`-h <host>` to connect to a specific host.

`-P <portNumber>` to connect to the host over the specified port. Port 902 is the default port the console uses with remote connections. For information about the port number, see Changing the Port Number for VMware Virtual Machine Console Connections in the *VMware GSX Server Administration Guide*.

`-u <username>` specifies the user name to use when you log on to a remote host.

`-w <password>` specifies the password to use when you log on to a remote host.

`-c "<path_to_virtualMachine>"` specifies the path to the virtual machine's configuration file to use to connect the console.

Startup Options on a Windows Host

The switches described above for Linux can also be used on a Windows host. The most convenient way to use the switches is to incorporate them into the command generated by a Windows shortcut.

Create the shortcut, right-click the shortcut, then click **Properties**. In the **Target** field, add any switches you want to use after the `vmware.exe` filename. For example,

```
"C:\Program Files\VMware\VMware GSX Server\vmware.exe -X
C:\Virtual Machines\Windows Server 2003\Windows Server 2003.vmx"
```

launches the specified Windows Server 2003 virtual machine, powers it on automatically and switches to full screen mode.

Be sure to enclose the entire command string in quotation marks.

Note: The configuration file has a `.vmx` extension by default. Pathnames on Windows use the backslash character (\). X toolkit options are not relevant on a Windows host.

Using Keyboard Shortcuts

If you prefer to work from the keyboard as much as possible, you may find the following keyboard shortcuts handy. If you have changed the Preferences setting for the hot-key combination, substitute your new setting for Ctrl-Alt as needed in the shortcuts listed here.

Shortcut	Action
Ctrl-B	Power on.
Ctrl-E	Power off.
Ctrl-R	Reset the power.
Ctrl-Z	Suspend.
Ctrl-N	Create a new virtual machine.
Ctrl-O	Open a virtual machine.
Ctrl-F4	Close the current virtual machine.
Ctrl-D	Edit the virtual machine's configuration.
Ctrl-G	Grab input from the keyboard and mouse.
Ctrl-P	Edit preferences. See Setting User Preferences for the VMware GSX Server Host in the *VMware GSX Server Administration Guide*.
Ctrl-Alt-Enter	Go to full screen mode.
Ctrl-Alt	Return to normal (windowed) mode.
Ctrl-Alt-Tab	Switch among open virtual machines while the mouse and keyboard input are grabbed.
Ctrl-Tab	Switch among open virtual machines while the mouse and keyboard input are not grabbed. VMware GSX Server must be the active application.
Ctrl-Shift-Tab	Switch among open virtual machines while the mouse and keyboard input are not grabbed. VMware GSX Server must be the active application.
Ctrl-Alt-Fx	Switch among open virtual machines while using full screen mode. Fx is a function key corresponding to the virtual machine you want to use. The key combination to use for a virtual machine is shown in the VMware GSX Server title bar when that virtual machine is active and in normal (windowed) mode.

CHAPTER 5

Preserving the State of a Virtual Machine

VMware GSX Server 3 offers two ways to preserve the state of a virtual machine. The following sections describe these features and help you understand which is appropriate in particular situations:

- Suspending and Resuming Virtual Machines on page 150
- Taking Snapshots on page 153

Suspending and Resuming Virtual Machines

The suspend and resume feature is most useful when you want to save the current state of your virtual machine, then pick up work later with the virtual machine in the same state it was when you stopped.

Once you resume and do additional work in the virtual machine, you cannot return to the state the virtual machine was in at the time you suspended unless you took a snapshot when you suspended.

To preserve the state of the virtual machine so you can return to the same state repeatedly, take a snapshot. For details, see Taking Snapshots on page 153.

The speed of the suspend and resume operations depends on how much data has changed while the virtual machine has been running. In general, the first suspend operation takes a bit longer than subsequent suspend operations do.

When you suspend a virtual machine, a file with a `.vmss` extension is created. This file contains the entire state of the virtual machine. When you resume the virtual machine, its state is restored from the `.vmss` file. The `.vmss` file cannot be used to resume a virtual machine again from the original suspended state.

Note: You should not change a configuration file after you suspend a virtual machine, since the virtual machine does not resume properly if the configuration file is inconsistent with the suspended virtual machine. Also, you should not move any physical (raw) disks that the virtual machine uses. If you do, the virtual machine cannot access its virtual disks when it resumes.

To suspend a virtual machine:

1. If your virtual machine is running in full screen mode, return to window mode by pressing the Ctrl-Alt key combination.
2. Click **Suspend** on the VMware Virtual Machine Console toolbar.
3. When GSX Server has completed suspending the virtual machine, it is safe to close the console.

   File > **Exit**

To resume a virtual machine that you have suspended:

1. Launch the VMware Virtual Machine Console and choose a virtual machine you have suspended.
2. Click **Resume** on the console toolbar.

Note that any applications you were running at the time you suspended the virtual machine are running and the content is the same as it was when you suspended the virtual machine.

You can suspend and resume a virtual machine with the management interface. See Changing a Virtual Machine's Power State from the Management Interface on page 120.

You can also set the configuration of each virtual machine so the file that stores information on the suspended state is saved in a location of your choice.

Setting the Suspended State File Directory

Recall that when a virtual machine is suspended, its state is written to a file with a `.vmss` extension. By default, the `.vmss` file is stored in the directory in which the virtual machine's configuration file (`.vmx`) resides. Similarly, when a virtual machine is being resumed, GSX Server looks for the `.vmss` file in the same directory.

To change the directory where the suspended state file for a virtual machine is stored, you must power off the virtual machine. You can specify this directory from the console's virtual machine settings editor or the VMware Management Interface.

Note: Changing the working directory also changes where you store the virtual machine's snapshot and redo log files.

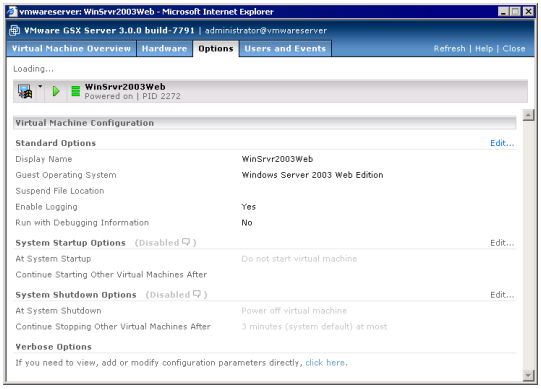
Setting the Suspended State File Directory from the Console

1. Connect to the virtual machine with a console, make sure the virtual machine is powered off, then choose **Edit virtual machine settings**.
2. On the **Options** tab, click **General**.
3. Under **Working Directory**, enter the name of a directory to use, or click **Browse** to select a directory.
4. Click **OK**.

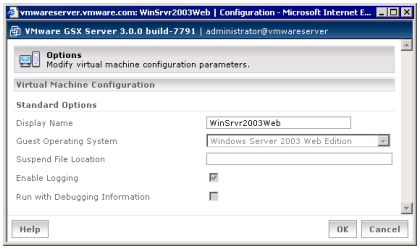
Setting the Suspended State File Directory from the Management Interface

1. Log on to the VMware Management Interface, then click the virtual machine menu icon ( ) for the virtual machine you want to change and choose **Configure Options**.

   The Options page for this virtual machine appears in a new browser window.



2. Click **Edit**. The Options page appears.



   For fastest suspend and restore operations, type the path to the appropriate directory in the **Suspend File Location** field. GSX Server automatically adds a suffix to the name of the suspended state file to ensure that one virtual machine does not overwrite the suspended state file of another.

3. Click **OK** to save your changes.

Taking Snapshots

The snapshot feature is most useful when you want to preserve the state of the virtual machine so you can return to the same state repeatedly.

To simply save the current state of your virtual machine, then pick up work later with the virtual machine in the same state it was when you stopped, suspend the virtual machine. For details, see Suspending and Resuming Virtual Machines on page 150.

You can take a snapshot of a virtual machine at any time and revert to that snapshot at any time. If the virtual machine is located on a Linux host, you should not take a snapshot while you are suspending the virtual machine; wait until the snapshot is completely saved, then take the snapshot.

You can take a snapshot while a virtual machine is powered on, powered off or suspended. A snapshot preserves the virtual machine just as it was when you took the snapshot — the state of the data on all the virtual machine's disks and whether the virtual machine was powered on, powered off or suspended.

Once you take a snapshot, the virtual machine starts saving any changes to the virtual machine to one or more redo-log files. The redo log can grow quite large as data is written to it. Be aware of how much disk space these logs consume. If you need to free up some disk space, you can remove a virtual machine's snapshot, which writes all the changes in the redo log to the virtual disk. For more information, see Snapshots and a Virtual Machine's Hard Disks on page 156 and Removing the Snapshot on page 157.

Note: If you are using a legacy virtual machine — for example, a virtual machine created under GSX Server 2 and not upgraded to use the new GSX Server 3 virtual hardware — you must power off the virtual machine before taking a snapshot. For information on upgrading the virtual hardware, see Upgrading VMware GSX Server in the *VMware GSX Server Administration Guide*. You also must power off the virtual machine before taking a snapshot if the virtual machine has multiple disks in different disk modes — for example, if you have a special purpose configuration that requires you to use an independent disk.

When you revert to a snapshot, you discard all changes made to the virtual machine since you took the snapshot. This includes any data written to the virtual disk and any changes to the virtual machine's configuration.

Similarly, if you take a snapshot of a virtual machine then modify the virtual machine's configuration, any changes you make to the configuration are not reflected in the snapshot. You need to take a new snapshot.

Use the **Snapshot** and **Revert** buttons on the console toolbar to take a snapshot and revert to it later.

You can take a new snapshot at any time. When you do so, you replace the previous snapshot. You can have only one active snapshot at a time.

Note: Taking a new snapshot when the virtual machine is powered off and a snapshot already exists can take a long time, as the original snapshot needs to be removed. While you are taking a new snapshot, other consoles may not be able to connect to the server host and the users trying to connect may see an error that the VMware Registration Service (`vmware-serverd`) is not running.

The following sections describe snapshots in greater detail.

- What Is Captured by a Snapshot? on page 154
- Ways of Using Snapshots on page 155
- Snapshots and a Virtual Machine's Hard Disks on page 156
- Snapshots and Other Activity in the Virtual Machine on page 156
- Settings for Snapshots on page 157
- Removing the Snapshot on page 157
- Snapshots and Legacy Disk Modes on page 158
- Snapshots and Repeatable Resume on page 158
- Snapshots and Legacy Virtual Machines on page 158

What Is Captured by a Snapshot?

A snapshot captures the entire state of the virtual machine at the time you take the snapshot. This includes:

- The state of all the virtual machine's disks.
- The contents of the virtual machine's memory.
- The virtual machine settings.

When you revert to the snapshot, you return all these items to the state they were in at the time you took the snapshot.

Note: In certain special purpose configurations, you may want to exclude one or more of the virtual machine's disks from the snapshot. To exclude a disk from the snapshot, choose **VM** > **Settings**, select the drive you want to exclude, then click **Advanced**. On the advanced settings screen, select **Independent**. You have the following options for an independent disk:

- Persistent — changes are immediately and permanently written to the disk. All changes to an independent disk in persistent mode remain, even when you revert to the snapshot.
- Nonpersistent — changes to the disk are discarded when you power off or revert to the snapshot.

Ways of Using Snapshots

The following examples illustrate the most common ways you can use a snapshot.

Always Saving Changes (There Is No Snapshot)

If you do not take a snapshot, your virtual machine runs the same way a physical computer does. All changes you make while you are working with a virtual machine are saved and you cannot return to an earlier state.

If you do not need to use the snapshot feature, it is best to run your virtual machine with no snapshot. This provides the best performance. To be sure a virtual machine has no snapshot, choose Snapshot > Remove Snapshot.

To configure the virtual machine not to use snapshots, choose VM > Settings > Options > Snapshot, then check Disable snapshots.

Another way to make sure the virtual machine doesn't use snapshots is to configure the virtual disk in independent mode. For more information, see Independent Disks on page 164.

Making Risky Changes

If you plan to make risky changes in a virtual machine (for example, testing new software or examining a virus), take a snapshot before you begin to make those risky changes. If you encounter a problem, click Revert on the console toolbar to return the virtual machine to its state at the time you took the snapshot.

If the first action you take causes no problems and you want to protect the virtual machine in its new state, you can take a new snapshot. You can have only one snapshot at a given time. When you take the new snapshot, you replace your previous snapshot and the contents of the previous snapshot are written to the virtual disk. You do not lose any data.

Starting a Virtual Machine Repeatedly in the Same State

You can configure the virtual machine to revert to the snapshot any time it is powered off. To do so, choose VM > Settings > Options > Snapshot. Under When powering off, select Revert to the snapshot. If you want the virtual machine to be suspended when you launch it, suspend the virtual machine before saving the snapshot. Similarly,

if you want the virtual machine to be powered on or powered off when you launch it, be sure it is powered on or powered off when you take the snapshot.

Snapshots and a Virtual Machine's Hard Disks

When a snapshot exists and the virtual machine saves data to disk, that data is written to a set of redo-log files. These files have `.REDO` as part of the filename and are stored in the virtual machine's working directory.

Redo-log files can grow quite large as newly saved data continues to accumulate in them until you take an action that affects the snapshot. Be aware of how much disk space these logs consume.

- Remove the snapshot — When you remove the snapshot, the changes accumulated in the redo-log files are written permanently to the base disks (either the virtual disk files or the physical disks, depending on your virtual machine's hard disk configuration). This is similar to committing changes to a disk in GSX Server 2.
- Revert to the snapshot — When you revert to the snapshot, the contents of the redo-log files are discarded. Any additional changes are, once again, accumulated in the redo-log files. This is similar to discarding changes to a disk in GSX Server 2.
- Take a snapshot — If you take a snapshot when the virtual machine already has a snapshot, changes stored in the redo-log files are written permanently to the base disk. Then any subsequent changes are, once again, accumulated in the redo-log files. Depending upon how large the redo-log file is, taking a new snapshot can take some time.

Snapshots and Other Activity in the Virtual Machine

When you take a snapshot, be aware of other activity going on in the virtual machine and the likely impact of reverting to the snapshot. In general, it is best to take the snapshot when no applications in the virtual machine are communicating with other computers.

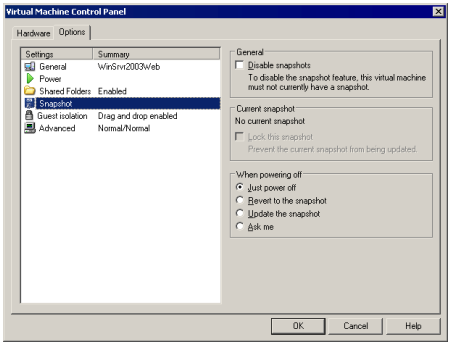
The potential for problems is greatest if the virtual machine is communicating with another computer, especially in a production environment.

Consider a case in which you take a snapshot while the virtual machine is downloading a file from a server on the network. After you take the snapshot, the virtual machine continues downloading the file, communicating its progress to the server. If you revert to the snapshot, communications between the virtual machine and the server are confused and the file transfer fails.

Or consider a case in which you take a snapshot while an application in the virtual machine is sending a transaction to a database on a separate machine. If you revert to the snapshot — especially if you revert after the transaction starts but before it has been committed — the database is likely to be confused.

Settings for Snapshots

You can specify what you want GSX Server to do with the snapshot any time the virtual machine is powered off. To do so, go to **VM** > **Settings** > **Options** > **Snapshot** and select one of the choices under **When powering off**.



Options when powering off include

- **Just power off** — leaves the snapshot as it is.
- **Revert to the snapshot** — reverts to the snapshot so the virtual machine always starts in the same state; reverting to the snapshot discards changes.
- **Update the snapshot**— takes a new snapshot of the virtual machine state as it was just before you powered off; this replaces the previous snapshot.
- **Ask me** — always asks what to do with the snapshot when you power off.

If the virtual machine has no snapshot, you can disable the snapshot feature by selecting **Disable snapshots**. If you have a snapshot and want to disable the snapshot feature, first go to the console window and choose **Snapshot** > **Remove Snapshot**. Then return to the virtual machine settings editor and select **Disable snapshots**.

To lock the snapshot so no new snapshot can be taken, select **Lock this snapshot**.

Removing the Snapshot

You can remove the snapshot any time the virtual machine is powered off. Removing the snapshot writes the contents of the snapshot to the virtual disk, it does not destroy any data in the virtual machine. Moving forward, any changes you make as

you run the virtual machine are written to the virtual disk. You cannot revert to a previous state because the snapshot no longer exists.

To remove the snapshot, shut down and power off the virtual machine. Then choose Snapshot > Remove Snapshot.

Note: Removing a snapshot when the virtual machine is powered off can take a long time, depending upon the size of the snapshot file. While you are removing the snapshot, other consoles may not be able to connect to the server host and the users trying to connect may see an error that the VMware Registration Service (`vmware-serverd`) is not running.

Snapshots and Legacy Disk Modes

If you are familiar with the disk modes used in earlier versions of GSX Server, you can use the snapshot to achieve equivalent results. If you want the equivalent of:

- Persistent mode — Do not take a snapshot.
- Undoable mode — Take a snapshot when you begin your working session. To discard all work done during the session, revert to the snapshot. To commit the work done during the session, take a new snapshot at the end of the working session. To keep the work done during a session without committing it, leave the original snapshot unchanged.
- Nonpersistent mode — Be sure the virtual machine is in the state you want it. Power off the virtual machine. Take a snapshot. Go to VM > Settings > Options > Snapshot. Under When powering off select Revert to snapshot.

Note: In earlier versions of GSX Server, disk modes had to be set individually for each disk. The snapshot introduced in GSX Server 3 applies by default to the entire virtual machine, including all disks attached to the virtual machine.

Snapshots and Repeatable Resume

The repeatable resume feature in GSX Server 2 allows you to resume a suspended virtual machine repeatedly in the same state. You can use the snapshot to accomplish the same thing. Run the virtual machine, be sure it is in the state you want it, then suspend it. Take a snapshot. Go to VM > Settings > Options > Snapshot. Under When powering off, select Revert to the snapshot.

Snapshots and Legacy Virtual Machines

If you are using a legacy virtual machine — for example, a virtual machine created under GSX Server 2 and not upgraded to use the new GSX Server 3 virtual hardware

— and you have disks in undoable or nonpersistent mode, you have a snapshot. If you have persistent disks, you have no snapshot. You have the following options:

- Persistent mode — You have no snapshot. You may take a snapshot any time the virtual machine is powered off.
- Undoable mode — You have a snapshot. You may update or remove the snapshot any time the virtual machine is powered off.
- Nonpersistent mode — You have a snapshot. In addition, in the virtual machine settings editor, the virtual machine is set to revert to the snapshot every time it is powered off. You may update or remove the snapshot any time the virtual machine is powered off. You may also change the settings in the virtual machine settings editor any time the virtual machine is powered off.

CHAPTER 6

Using Disks in a Virtual Machine

The following sections provide information on configuring your virtual machine's hard disk storage so the virtual disk best meets your needs:

- Configuring Hard Disk Storage in a Virtual Machine on page 162
- Configuring Optical and Floppy Drives on page 171
- Adding Drives to a Virtual Machine on page 177
- Using VMware Virtual Disk Manager on page 191
- Configuring a Dual-Boot Computer for Use with a Virtual Machine on page 198
- Installing an Operating System onto a Physical Partition from a Virtual Machine on page 220
- Disk Performance in Windows NT Guests on Multiprocessor Hosts on page 225

Configuring Hard Disk Storage in a Virtual Machine

Like a physical computer, a VMware GSX Server virtual machine stores its operating system, programs and data files on one or more hard disks. Unlike a physical computer, GSX Server gives you options for undoing changes to the virtual machine's hard disk.

The New Virtual Machine Wizard creates a virtual machine with one disk drive. You can use the virtual machine settings editor (**VM** > **Settings**) to add more disk drives to your virtual machine, to remove disk drives from your virtual machine or to change certain settings for the existing disk drives.

The following sections describe the choices you can make in setting up hard disk storage for your virtual machine.

- Disk Types: Virtual and Physical on page 162
- Additional Information about Disk, Redo-Log, Snapshot and Lock Files on page 165
- Defragmenting and Shrinking Virtual Disks on page 168

Disk Types: Virtual and Physical

In the most common configurations, GSX Server creates virtual hard disks, which are made up of files that are typically stored on your host computer's hard disk. In some circumstances, you may need to give your virtual machine direct access to a physical hard drive on your host computer — using the disk type referred to as a physical disk.

Virtual Disk

A virtual disk is a file or set of files that appears as a physical disk drive to a guest operating system. The files can be on the host machine or on a remote computer. When you configure a virtual machine with a virtual disk, you can install a new operating system onto the virtual disk without repartitioning a physical disk or rebooting the host.

IDE virtual disks can be as large as 128GB. SCSI virtual disks can be as large as 256GB. Depending on the size of the virtual disk and the host operating system, GSX Server creates one or more files to hold each virtual disk.

By default, the virtual disk is configured so all the disk space is allocated at the time the virtual disk is created. This type of virtual disk is known as a **preallocated disk**. A preallocated disk provides enhanced performance and is useful if you are running performance-sensitive applications in the virtual machine. Virtual disks created in this

way are similar to the experimental plain disks that could be created under earlier versions of GSX Server.

A virtual disk that is not preallocated is known as a **growable disk**. A growable virtual disk's files start out small and grow to their maximum size as needed. The main advantage of this approach is the smaller file size. Smaller files require less storage space and are easier to move if you want to move the virtual machine to a new location. You can shrink this type of virtual disk. However, it takes longer to write data to a disk configured in this way.

Virtual disks can be set up as IDE disks for any guest operating system. They can be set up as SCSI disks for any guest operating system that has a driver for the BusLogic SCSI adapter used in a GSX Server virtual machine.

Note: To use SCSI disks in a Windows XP or Windows Server 2003 virtual machine, you need a special SCSI driver available from the download section of the VMware Web site at *www.vmware.com/download*. Follow the instructions on the Web site to use the driver with a fresh installation of Windows XP or Windows Server 2003.

A virtual disk of either type can be stored on either type of physical hard disk. That is, the files that make up an IDE virtual disk can be stored on either an IDE hard disk or a SCSI hard disk. So can the files that make up a SCSI virtual disk. They can also be stored on other types of fast-access storage media, such as DVD-ROM or CD-ROM discs. For information about running virtual machines from DVD-ROM or CD-ROM, see Running Virtual Machines from DVD-ROM or CD-ROM Discs on page 130.

A key advantage of virtual disks is their portability. Because the virtual disks are stored as files on the host machine or a remote computer, you can move them easily to a new location on the same computer or to a different computer. You can also use GSX Server on a Windows host to create virtual disks, then move them to a Linux computer and use them under GSX Server for Linux — or vice versa. For information about moving virtual disks, see Moving and Sharing Virtual Machines in the *VMware GSX Server Administration Guide*.

Physical (Raw) Disk

A physical disk directly accesses an existing local disk or partition. You can use physical disks if you want GSX Server to run one or more guest operating systems from existing disk partitions. Physical disks may be set up on both IDE and SCSI devices. At this time, however, booting from an operating system already set up on an existing SCSI disk or partition is not supported.

The most common use of a physical disk is to convert a dual-boot or multiple-boot machine so one or more of the existing operating systems can be run inside a virtual machine.

Physical disks may be set up on both IDE and SCSI devices. Physical disks can be as large as 128GB when configured as IDE or 256GB when configured as SCSI.

Caution: You cannot use a physical disk that is stored on a SAN. You must use a disk or a partition on the GSX Server host.

Caution: If you run an operating system natively on the host computer, then switch to running it inside a virtual machine, the change is like pulling the hard drive out of one computer and installing it in a second computer with a different motherboard and other hardware. You need to prepare carefully for such a switch. The specific steps you need to take depend on the operating system you want to use inside the virtual machine.

You can create a new virtual machine that uses a physical disk instead of a virtual disk. For details, see Installing an Operating System onto a Physical Partition from a Virtual Machine on page 220. In most cases, however, it is better to use a virtual disk.

Only advanced users should attempt physical disk configurations.

Note: You should not use a physical disk to share files between host and guest operating systems. It is not safe to make the same partition visible to both host and guest. You can cause data corruption if you do this.

In older VMware products, physical disks were called raw disks.

Independent Disks

Independent disks add a layer of control and complexity to your virtual disks. You configure virtual disks in independent mode for certain special purpose configurations.

For example, you may want to run a virtual machine that uses a virtual disk stored on DVD-ROM or CD-ROM. For more information, see Running Virtual Machines from DVD-ROM or CD-ROM Discs on page 130.

Or, you may want to exclude one or more virtual disks from a virtual machine's snapshot. For more information about snapshots, see Taking Snapshots on page 153.

To configure a disk as an independent disk, choose **VM** > **Settings**, select the virtual disk in question, then click **Advanced**. On the advanced settings screen, select **Independent**, then the mode for the disk. You have the following options for an independent disk:

- **Persistent** — changes are immediately and permanently written to the disk. All changes to an independent disk in persistent mode remain, even when you revert to the snapshot.
- **Nonpersistent** — changes to the disk are discarded when you power off or revert to the snapshot. Choose this option if you want to run a virtual machine where the virtual disk is stored on a DVD-ROM or CD-ROM, or if you want to lose any changes made to the virtual disk since the snapshot was taken when you revert to the snapshot.

Additional Information about Disk, Redo-Log, Snapshot and Lock Files

This section provides more information about various virtual machine files.

Disk Files

The virtual machine settings editor (**VM** > **Settings**) allows you to choose the disk files for a virtual machine.

You may want to choose a file other than the one created by the New Virtual Machine Wizard if you are using a virtual disk that you created in a different location or if you are moving the automatically created disk files to a new location.

The disk files for a virtual disk store the information that you write to a virtual machine's hard disk — the operating system, the program files and the data files. The virtual disk files have a `.vmdk` extension.

A virtual disk is made up of one or more `.vmdk` files.

On Windows hosts, each virtual disk is contained in one file by default. You may, as an option, configure the virtual disk to use a set of files limited to 2GB per file. Use this option if you plan to move the virtual disk to a file system that does not support files larger than 2GB.

You must set this option at the time you create the virtual disk.

If you are setting up a new virtual machine, follow the custom path in the New Virtual Machine Wizard. In the screen that allows you to specify the virtual disk's capacity, select **Split disk into 2GB files**.

If you are adding a virtual disk to an existing virtual machine, follow the steps in the Add Hardware Wizard. In the screen that allows you to specify the virtual disk's capacity, select **Split disk into 2GB files**.

When a disk is split into multiple files, larger virtual disks have more `.vmdk` files.

The first `.vmdk` file for each disk is small and contains pointers to the other files that make up the virtual disk. The other `.vmdk` files contain data stored by your virtual machine and use a small amount of space for virtual machine overhead.

By default, all disk space is allocated when you create the virtual disk. A preallocated virtual disk has fixed file sizes, and most of the files are 2GB. As mentioned above, the first file is small. The last file in the series may also be smaller than 2GB.

If you chose to not allocate the space in advance, the `.vmdk` files grow as data is added, to a maximum of 2GB each — except for the first file in the set, which remains small.

The virtual machine settings editor shows the name of the first file in the set — the one that contains pointers to the other files in the set. The other files used for that disk are automatically given names based on the name of the first file.

For example, a Windows 2000 Server virtual machine using the default configuration, with files that grow as needed, stores the disk in files named `Windows 2000 Server.vmdk`, `Windows 2000 Server-s001.vmdk`, `Windows 2000 Server-s002.vmdk` and so on.

If the disk space is allocated in advance and the virtual disk is split into 2GB files, the names are similar, except that they include an `f` instead of an `s` — for example, `Windows 2000 Server-f001.vmdk`. If the disk is not split into 2GB files, the virtual machine stores the disk in two files, named `Windows 2000 Server.vmdk` and `Windows 2000 Server-flat.vmdk`.

If you are using a physical disk, the `.vmdk` file stores information about the physical disk or partition used by the virtual machine.

Redo-Log and Snapshot Files

Redo-log files save blocks that the virtual machine modifies while it is running. The redo-log file for a disk in independent-nonpersistent mode is not saved when the virtual machine is powered off or reset, while the redo-log file for a disk with a snapshot is saved. This file is known as the redo log.

The redo-log file for a virtual disk called `vm` is called `vm.vmdk.REDO`. If the virtual disk is split into 2GB files, the disk files are named `vm.vmdk`, `vm-02.vmdk`, `vm-03.vmdk` and so on; its redo-log files are called `vm.vmdk.REDO`, `vm-02.vmdk.REDO`, `vm-03.vmdk.REDO` and so on.

When you take a snapshot of a virtual machine called `vm`, GSX Server stores the snapshot in a file is called `vm.vmsn`. For more information about snapshots, see Taking Snapshots on page 153.

You can choose the location where the redo-log and snapshot files are stored. By default, the files are stored in the same directory as the virtual disk (`.vmdk`) file.

By default, redo-log files for physical disks are located in the same directory as the virtual machine configuration file (`.vmx`).

You can change the location of the redo-log and snapshot files in the virtual machine settings editor. With the virtual machine powered off, choose **VM** > **Settings**. Click the **Options** tab, select **General**, then under **Working directory**, type in or browse to the folder in which the redo log or snapshot should be stored.

You may choose to locate these files in a different directory to increase available space or improve performance. For best performance, the log files for a virtual machine should be on a local hard drive on the host computer.

Lock Files

A running virtual machine creates lock files to prevent consistency problems on virtual disks. If the virtual machine did not use locks, multiple virtual machines might read and write to the disk, causing data corruption.

Lock files are always created in the same directory as the `.vmdk` file.

The locking methods used by GSX Server on Windows and Linux hosts are different, so files shared between them are not fully protected. If you use a common file repository that provides files to users on both Windows and Linux hosts, be sure that each virtual machine is run by only one user at a time.

There is a way to work around the lock file so that multiple virtual machines can access it — by using SCSI reservation. This is typically done in conjunction with a high-availability configuration, such as clustering. For more information about this, see High-Availability Configurations with VMware GSX Server in the *VMware GSX Server Administration Guide*.

When a virtual machine is powered off, it removes the lock files it created. If it cannot remove the lock, a stale lock file is left protecting the `.vmdk` file. For example, if the host machine crashes before the virtual machine has a chance to remove its lock file, a stale lock remains.

If a stale lock file remains when the virtual machine is started again, the virtual machine tries to remove the stale lock. To make sure that no virtual machine could be using the lock file, the virtual machine checks the lock file to see if

1. The lock was created on the same host where the virtual machine is running.
2. The process that created the lock is not running.

If those two conditions are true, the virtual machine can safely remove the stale lock. If either of those conditions is not true, a dialog box appears, warning you that the virtual machine cannot be powered on. If you are sure it is safe to do so, you may delete the lock files manually. On Windows hosts, the filenames of the lock files end in `.lck`. On Linux hosts, the filenames of the lock files end in `.WRITELOCK`.

Physical disk partitions are also protected by locks. However, the host operating system is not aware of this locking convention and thus does not respect it. For this reason, VMware strongly recommends that the physical disk for a virtual machine not be installed on the same physical disk as the host operating system.

Defragmenting and Shrinking Virtual Disks

If you have a virtual disk that grows as data is added, you can defragment and shrink it as described in this section. If you allocated all the space for your virtual disk at the time you created it, you cannot defragment and shrink it.

Defragmenting Virtual Disks

Defragmenting disks rearranges files, programs and unused space on the virtual disk so that programs run faster and files open more quickly. Defragmenting does not reclaim unused space on a virtual disk; to reclaim unused space, shrink the disk.

For best disk performance, you can take the following three actions, in the order listed:

1. Run a disk defragmentation utility inside the virtual machine.
2. Power off the virtual machine, then defragment its virtual disks from the virtual machine settings editor (**VM** > **Settings**). Select the virtual disk you want to defragment, then click **Defragment**.

   Note: This capability works only with virtual disks, not with raw or plain disks (plain disks are a feature of older VMware products).
3. Run a disk defragmentation utility on the host computer.

Defragmenting disks may take considerable time.

Note: The defragmentation process requires free working space on the host computer's disk. If your virtual disk is contained in a single file, for example, you need free space equal to the size of the virtual disk file. Other virtual disk configurations require less free space.

Shrinking Virtual Disks

Shrinking a virtual disk reclaims unused space in the virtual disk. If there is empty space in the disk, this process reduces the amount of space the virtual disk occupies on the host drive. You cannot shrink preallocated virtual disks or physical disks.

Shrinking virtual disks is a convenient way to convert a virtual disk to the format supported by GSX Server 3. Virtual disks created in the new format cannot be recognized by earlier VMware products except for VMware Workstation 3.0 and higher.

The virtual disks to be shrunk must not be booted as independent disks. You can change the mode of a virtual disk before the virtual machine is powered on. See Independent Disks on page 164.

Shrinking requires free disk space on the host equal to the size of the virtual disk being shrunk.

Shrinking a disk is a two-step process: in the first step, called wiping, VMware Tools reclaims all unused portions of disk partitions (such as deleted files) and prepares them for shrinking. This allows for the maximum shrink possible. Wiping takes place in the guest operating system.

The shrink process itself is the second step, and it takes place outside the virtual machine. GSX Server reduces the size of the disk based on the disk space reclaimed by the wipe process. This step occurs after the wipe finishes preparing the disk for shrinking.

When a virtual machine is powered on, you shrink its virtual disks from the VMware Tools control panel. You cannot shrink virtual disks if a snapshot exists. To remove an existing snapshot, choose **Snapshot** > **Remove Snapshot**.

In a Linux or FreeBSD guest operating system, to prepare virtual disks for shrinking, you should run VMware Tools as the root user. This way, you ensure the whole virtual disk is shrunk. If you shrink disks as a non-root user, you cannot wipe the parts of the virtual disk that require root-level permissions.

1. To launch the control panel in a Windows guest, double-click the VMware Tools icon in the system tray or choose **Start** > **Settings** > **Control Panel**, then double-click **VMware Tools**.

   To launch the control panel in a Linux or FreeBSD guest, become root (`su -`), then run `vmware-toolbox`.

2. Click the **Shrink** tab.

3. Select the virtual disks you want to shrink, then click **Prepare to Shrink**.

   Note: If you deselect some of the partitions to wipe, the whole disk is still shrunk. However, those partitions are not prepared for shrinking, and the shrink process does not reduce the size of the virtual disk as much as it could otherwise.

4. When VMware Tools finishes wiping the selected disk partitions, you are prompted to begin shrinking the disks.

Shrinking disks may take considerable time.

In some configurations, it is not possible to shrink virtual disks. If your virtual machine uses such a configuration, the **Shrink** tab displays information explaining why you cannot shrink your virtual disks. For example, you cannot shrink a virtual disk if:

- You preallocated disk space when you created the disk, which is the default option for both typical and custom virtual machine creation paths.
- The virtual machine has a snapshot.
- The virtual machine contains physical (raw) disks.
- The virtual disk is not an independent disk in persistent mode. For more information, see Independent Disks on page 164.
- The virtual disk is stored on a CD-ROM.

Configuring Optical and Floppy Drives

The following sections describe how to configure your virtual machine's optical (DVD-ROM and CD-ROM) and floppy drives. You can use the physical device or point the virtual machine to an ISO or floppy image file.

- Configuring Virtual DVD-ROM and CD-ROM Drives on page 171
- Configuring Virtual Floppy Drives on page 175

Configuring Virtual DVD-ROM and CD-ROM Drives

Each virtual machine can access a physical DVD-ROM or CD-ROM drive on the GSX Server host or an ISO image file.

Multiple virtual machines can connect to the DVD-ROM or CD-ROM drive on the GSX Server host at the same time, unless a virtual machine is configured to exclusively use the drive. For information about exclusive use of the optical drive, see Exclusively Using the DVD-ROM or CD-ROM Drive on page 173.

You configure virtual DVD-ROM and CD-ROM drives from the virtual machine settings editor (**VM** > **Settings**) or the VMware Management Interface.

Options you can configure include choosing the device node for the guest, using legacy emulation mode, using the optical drive on a client instead of the GSX Server host and exclusively using the DVD-ROM or CD-ROM drive.

Configuring a Virtual Machine's DVD-ROM or CD-ROM Drive from the Console

To configure a virtual machine's DVD-ROM or CD-ROM drive, complete the following steps.

1. Connect to the virtual machine with the VMware Virtual Machine Console.
2. Open the virtual machine settings editor. Choose **VM** > **Settings**.
3. On the Hardware tab, select the CD-ROM drive. You can make any of the following changes.
   - Choosing a Device Node for the DVD-ROM or CD-ROM Drive on page 172
   - Using Legacy Emulation for DVD-ROM and CD-ROM Drives on page 172
   - Exclusively Using the DVD-ROM or CD-ROM Drive on page 173
   - Using the DVD-ROM or CD-ROM Drive on a Client on page 173
4. Click **OK** to save your changes and close the virtual machine settings editor.

Choosing a Device Node for the DVD-ROM or CD-ROM Drive

Like a virtual disk, the virtual machine's DVD-ROM or CD-ROM drive can be associated with a specific SCSI or IDE device node.

The type of device does not have to match the type of device on the host, so if your GSX Server host has an IDE CD-ROM drive, you can still configure your virtual machine with a SCSI CD-ROM drive.

If you want to do more than read data from the drive — for example, burn CD-ROMs — you should match the bus types. So if your host has an IDE CD-ROM drive, configure the virtual CD-ROM drive on an IDE device node.

However, if you want to boot from a virtual CD-ROM drive, you must configure the drive as an IDE device.

Using Legacy Emulation for DVD-ROM and CD-ROM Drives

The virtual machine settings editor provides a **Legacy emulation** option for DVD-ROM and CD-ROM drives attached to the virtual machine.

On Windows hosts, this option is deselected by default.

On Linux hosts with IDE drives, the default setting for this option depends on whether the `ide-scsi` module is loaded in your kernel. The `ide-scsi` module must be loaded — or you must be using a physical SCSI drive — if you want to connect to the DVD-ROM or CD-ROM drive in raw mode.

If you encounter problems using your DVD-ROM or CD-ROM drive, try selecting **Legacy emulation**.

Note that in legacy emulation mode, you can read from data discs in the DVD-ROM or CD-ROM drive, but some other functions are not available. For example, you cannot read from multisession discs if your DVD-ROM or CD-ROM drive is configured for legacy mode. You cannot burn CD-ROMs either.

When **Legacy emulation** is deselected, the guest operating system communicates directly with the drive. This direct communication enables capabilities that are not possible in legacy emulation mode, such as using CD and DVD writers to burn discs, reading multisession CDs, performing digital audio extraction and viewing video.

However, in some cases, the DVD-ROM or CD-ROM drive may not work correctly when the guest operating system is communicating directly with the drive. In addition, certain drives and their drivers do not work correctly in raw mode. Selecting **Legacy emulation** is a way to work around these problems.

Exclusively Using the DVD-ROM or CD-ROM Drive

You can prevent other virtual machines and the host from using the DVD-ROM or CD-ROM drive until either you disconnect it from this virtual machine or you power off or suspend the virtual machine. In the virtual machine settings editor, check **Connect exclusively to this virtual machine**.

Using the DVD-ROM or CD-ROM Drive on a Client

When you use the VMware Virtual Machine Console on a remote client to connect to a virtual machine, you have the option of using the optical drive on the client machine instead of the drive on the GSX Server host. This is a convenient way of installing software remotely if you do not have access to the host.

To use a client machine's DVD-ROM or CD-ROM drive, make sure you are using the physical drive. Next to **Location**, select **Client**.

All virtual machine settings — like using legacy emulation and exclusive connections — apply, except that a CD-ROM drive on a client cannot start connected.

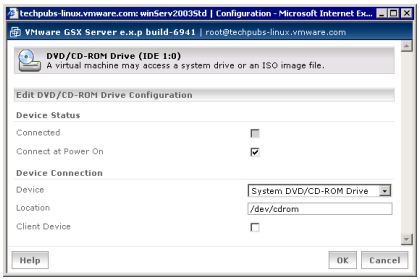
If you want to boot the virtual machine from the DVD-ROM or CD-ROM drive in a client system, complete the following steps.

1. When you first begin booting the guest operating system, press the Esc key. A boot menu appears.
2. Choose one of the following.
   - Open the virtual machine settings editor (**VM** > **Settings**) and select the CD-ROM drive. Select **Use physical drive**, then next to **Location**, select **Client**.
   - In the VMware Management Interface, edit the DVD-ROM or CD-ROM drive in the virtual machine's hardware page. Select the physical drive in the **Device** list, then check the **Client Device** check box.
3. Return to the boot menu in the guest operating system. Select the CD-ROM drive, then press Enter to boot the virtual machine from the CD-ROM drive of the client on which you are running the VMware Virtual Machine Console.

Configuring a Virtual Machine's DVD-ROM or CD-ROM Drive from the Management Interface

To configure a virtual machine's DVD-ROM or CD-ROM drive, complete the following steps.

1. In the Hardware page, under **DVD/CD-ROM Drive**, click **Edit**. The DVD/CD-ROM Drive page appears.



2. To connect this virtual machine to the host's DVD/CD-ROM drive, check **Connected**.
3. To connect this virtual machine to the host's DVD/CD-ROM drive when the virtual machine is powered on, check **Connect at Power On**.
4. Specify whether to connect to the host's DVD/CD-ROM drive or to an ISO image. In the **Device** list, select **System DVD/CD-ROM Drive** or **ISO Image**.
5. Enter the location of the drive or ISO image in the **Location** field. For example, the host's CD-ROM drive could be `D:` or `/dev/cdrom`.
6. To use the client machine's DVD-ROM or CD-ROM drive, make sure you are using the physical drive, then check the **Client Device** check box For more information, see Using the DVD-ROM or CD-ROM Drive on a Client on page 173.
7. Click **OK** to save your changes and close the window.

Configuring Virtual Floppy Drives

Each virtual machine can access a physical floppy drive on the GSX Server host or a floppy image file.

Only one virtual machine can connect to the floppy drive on the server at a time.

You configure virtual floppy drives from the virtual machine settings editor or the VMware Management Interface.

Configuring a Virtual Machine's Floppy Drive from the Console

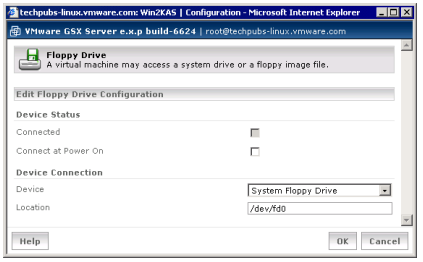
To configure a virtual machine's floppy drive, complete the following steps.

1. Connect to the virtual machine with the VMware Virtual Machine Console.
2. Open the virtual machine settings editor. Choose **VM** > **Settings**.
3. On the Hardware tab, select the floppy drive.
4. To connect this virtual machine to the floppy drive when the virtual machine is powered on, check **Connect at Power On**.
5. Specify whether to connect to the host's floppy drive or to a floppy image. Select **Use physical drive**, then choose the drive from the list. Or select **Use floppy Image**, then create a new or browse to an existing floppy image.
6. Click **OK** to save your changes and close the virtual machine settings editor.

Configuring a Virtual Machine's Floppy Drive from the Management Interface

To configure the virtual machine's floppy drive, complete the following steps.

1. In the Hardware page, under **Floppy Drive**, click **Edit**. The Floppy Drive page appears.



2. To connect this virtual machine to the floppy drive, check **Connected**.
3. To connect this virtual machine to the floppy drive when the virtual machine is powered on, check **Connect at Power On**.

4. Specify whether to connect to the host's floppy drive or to a floppy image. In the **Device** list, select **System Floppy Drive** or **Floppy Image**.
5. Enter the location of the drive or floppy image in the **Location** field. For example, the host's floppy drive could be `A:` or `/dev/fd0`.
6. Click **OK** to save your changes and close the window.

Adding Drives to a Virtual Machine

A GSX Server virtual machine can use up to four IDE devices and up to seven SCSI devices. Any of these devices can be a virtual hard disk or DVD or CD-ROM drive. A virtual machine can read data from a DVD-ROM disc. GSX Server does not support playing DVD movies in a virtual machine.

Many other SCSI devices can be connected to a virtual machine using the host operating system's generic SCSI driver. For details on connecting these devices, see Connecting to a Generic SCSI Device on page 325.

The following sections describe how to add virtual disks, physical disks, DVD-ROM/CD-ROM drives and floppy drives to virtual machines. In addition, you can connect CD-ROM and floppy drives to disk image files.

- Adding Virtual Disks to a Virtual Machine on page 177
- Adding Physical Disks to a Virtual Machine on page 184
- Adding DVD-ROM or CD-ROM Drives to a Virtual Machine on page 187
- Adding Floppy Drives to a Virtual Machine on page 189

Adding Virtual Disks to a Virtual Machine

Virtual disks are stored as files on the host computer or on a network file server. It does not matter whether the disk that holds the files is IDE or SCSI. A virtual IDE drive can be stored on an IDE drive or on a SCSI drive. So can a virtual SCSI drive.

Use the virtual machine settings editor to add a new virtual disk to your virtual machine. The virtual machine should be powered off before you begin. If it is not, shut down the guest operating system normally, then click **Power Off** on the VMware Virtual Machine Console toolbar.

Note: If you have a Windows NT 4.0 guest with a SCSI virtual disk, you cannot add both an additional SCSI disk and an IDE disk to the configuration.

Adding a New Virtual Disk from the Console

1. Open the virtual machine settings editor (**VM** > **Settings**) and click **Add**. The Add Hardware Wizard guides you through the steps to create your virtual disk. Click **Next** to start configuring the virtual disk.
2. Click **Hard Disk**, then click **Next**.
3. Select **Create a new virtual disk**, then click **Next**.
4. Choose the type of virtual disk. The wizard recommends whether to use SCSI or IDE, based on the guest operating system installed in the virtual machine.

5. Set the capacity for the new virtual disk.

   You can set a size between 0.1GB (100MB) and 256GB for a SCSI virtual disk or 128GB for an IDE virtual disk. The default is 4GB.

   By default, **Allocate all disk space now** is checked.

   Allocating all the space at the time you create the virtual disk gives somewhat better performance, but it requires as much disk space as the size you specify for the virtual disk.

   A preallocated virtual disk is useful for clustering virtual machines. For more information about clustering, see High-Availability Configurations with VMware GSX Server in the *VMware GSX Server Administration Guide*.

   If you deselect this option, the virtual disk's files start small and grow as needed, but they can never grow larger than the size you set here.

   You may also specify whether you want the virtual disk created as one large file or split into a set of 2GB files. If you want to split the disk, select **Split disk into 2GB files**. You should split the virtual disk if it is stored on a FAT32 file system or on a file system that cannot support files larger than 2GB, such as FAT16.

6. Accept the default filename and location for the virtual disk file, or change it if you want to use a different name or location. To find a different folder, click **Browse**.

   If you want to specify a device node for your virtual disk, click **Advanced**.

   On the advanced settings screen, you can also specify a disk mode. This is useful in certain special-purpose configurations in which you want to exclude disks from the snapshot. For more information on snapshots, see Taking Snapshots on page 153. You can choose between a normal disk and an independent disk.

   Normal disks are included in snapshots. This is the default setting for a new disk.

   Independent disks are not included in snapshots. If you select **Independent**, you must further select one of the following modes:

   - **Persistent** — changes are immediately and permanently written to the disk.
   - **Nonpersistent** — changes to the disk are discarded when you power off or revert to the snapshot.

   When you have set the filename and location you want to use and have made any selections you want to make on the advanced settings screen, click **Finish**.

7. The wizard creates the new virtual disk. It appears to your guest operating system as a new, blank hard disk. Use the guest operating system's tools to partition and format the new drive.

Adding an Existing Virtual Disk from the Console

1. Open the virtual machine settings editor (**VM** > **Settings**) and click **Add**. The Add Hardware Wizard guides you through the steps to create your virtual disk. Click **Next** to start configuring the virtual disk.
2. Click **Hard Disk**, then click **Next**.
3. Select **Use an existing virtual disk**, then click **Next**.
4. Click **Browse**, then browse to the virtual disk (`.vmdk`) you want to use.
5. To associate the virtual disk with a specific device node, click **Advanced** and select the device node in the **Virtual device node** list.

   On the advanced settings screen, you can also specify a disk mode. This is useful in certain special-purpose configurations in which you want to exclude disks from the snapshot. For more information on the snapshot feature, see Taking Snapshots on page 153. You can choose between a normal disk and an independent disk.

   Normal disks are included in snapshots. This is the default setting for a new disk.

   Independent disks are not included in snapshots. If you select **Independent**, you must further select one of the following modes:

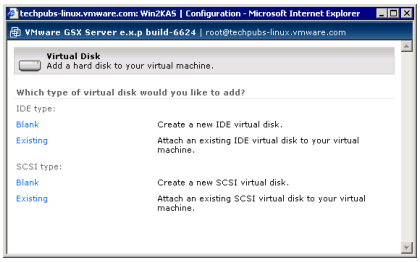
   - **Persistent** — changes are immediately and permanently written to the disk.
   - **Nonpersistent** — changes to the disk are discarded when you power off or revert to the snapshot.

   When you have set the filename and location you want to use and have made any selections you want to make on the advanced settings screen, click **Finish**. The wizard adds the virtual disk to the virtual machine.

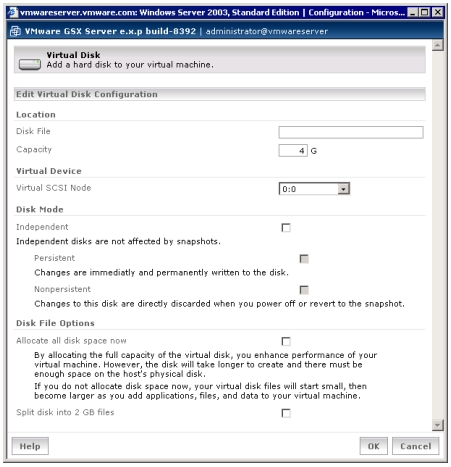
Adding a New Virtual Disk from the Management Interface

To add a new virtual disk to a virtual machine, make sure the virtual machine is powered off, then complete the following steps.

1. On the Hardware page, click **Add Device**. The Add Device Wizard starts.
2. Click **Hard Disk**. The virtual disk type page appears.


techpubs-linux.vmware.com: Win2KAS | Configuration - Microsoft Internet Explorer
VMware GSX Server e.x.p build-6624 | root@techpubs-linux.vmware.com
Virtual Disk
Add a hard disk to your virtual machine.
Which type of virtual disk would you like to add?
IDE type:
Blank
Create a new IDE virtual disk.
Existing
Attach an existing IDE virtual disk to your virtual machine.
SCSI type:
Blank
Create a new SCSI virtual disk.
Existing
Attach an existing SCSI virtual disk to your virtual machine.

3. To create a new virtual disk, decide whether you want the disk to be IDE or SCSI. The wizard suggests the recommended type. Then under **IDE type** or **SCSI type**, click **Blank**. The Virtual Disk Configuration page appears.


vmwareserver.vmware.com: Windows Server 2003, Standard Edition | Configuration - Micros...
VMware GSX Server e.x.p build-8392 | administrator@vmwareserver
Virtual Disk
Add a hard disk to your virtual machine.
Edit Virtual Disk Configuration
Location
Disk File
Capacity
4 G
Virtual Device
Virtual SCSI Node
0:0
Disk Mode
Independent
Independent disks are not affected by snapshots.
Persistent
Changes are immediatly and permanently written to the disk.
Nonpersistent
Changes to this disk are directly discarded when you power off or revert to the snapshot.
Disk File Options
Allocate all disk space now
By allocating the full capacity of the virtual disk, you enhance performance of your virtual machine. However, the disk will take longer to create and there must be enough space on the host's physical disk.
If you do not allocate disk space now, your virtual disk files will start small, then become larger as you add applications, files, and data to your virtual machine.
Split disk into 2 GB files
Help
OK
Cancel

4. In the **Disk File** field, enter the location and name of the virtual disk. Make sure the virtual disk has a `.vmdk` extension.
5. In the **Capacity** field, specify the size of the virtual disk in megabytes (MB). By default, GSX Server preallocates the space for the virtual disk when you create it.

   The virtual disk can be as small as 0.1GB (100MB). A SCSI virtual disk can be as large as 256GB; an IDE virtual disk can be as large as 128GB. The default is 4GB.

6. Specify the virtual device node in the **Virtual IDE Node** or **Virtual SCSI Node** list, as appropriate.
7. Decide if you want to make this virtual disk an independent disk. Under **Disk Mode**, check **Independent**, then check **Persistent** or **Nonpersistent**.

   Independent disks are not included in snapshots.

   Note: The independent disk option should be used only by advanced users who need it for special-purpose configurations.

   You have two options for an independent disk. You can make the disk **Persistent**, which means that changes are immediately and permanently written to the disk. Or you can make the disk **Nonpersistent**, which means that changes to the disk are discarded when you power off or reset the virtual machine.
8. Decide if you want to preallocate the virtual disk space. Allocating all the space at the time you create the virtual disk gives somewhat better performance and ensures you do not run out of disk space on the host, but it requires as much disk space as the size you specify for the virtual disk. You cannot shrink a preallocated disk. To preallocate the virtual disk, select **Allocate all disk space now**.

   A preallocated virtual disk is needed for clustering virtual machines. For more information about clustering, see High-Availability Configurations with VMware GSX Server in the *VMware GSX Server Administration Guide*.

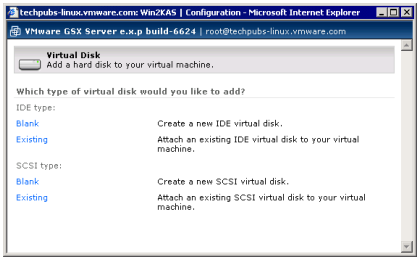
   If you do not preallocate the disk, the virtual disk's files start small and grow as needed, but they can never grow larger than the size you set here.

   You may also specify whether you want the virtual disk created as one large file or split into a set of 2GB files. You should split the virtual disk if it is stored on a FAT32 file system or a file system that cannot support files larger than 2GB, such as FAT16. To do this, check **Split into 2GB files**.
9. Click **OK** to add the virtual disk.

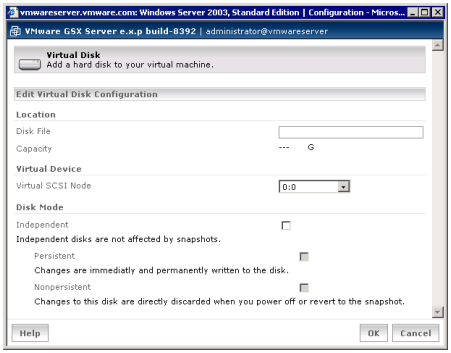
Adding an Existing Virtual Disk from the Management Interface

To add an existing virtual disk to a virtual machine, make sure the virtual machine is powered off, then complete the following steps.

1. On the Hardware page, click **Add Device**. The Add Device Wizard starts.
2. Click **Hard Disk**. The virtual disk type page appears.


techpubs-linux.vmware.com: Win2KAS | Configuration - Microsoft Internet Explorer
VMware GSX Server e.x.p build-6624 | root@techpubs-linux.vmware.com
Virtual Disk
Add a hard disk to your virtual machine.
Which type of virtual disk would you like to add?
IDE type:
Blank
Create a new IDE virtual disk.
Existing
Attach an existing IDE virtual disk to your virtual machine.
SCSI type:
Blank
Create a new SCSI virtual disk.
Existing
Attach an existing SCSI virtual disk to your virtual machine.

3. To add an existing virtual disk, choose the correct disk type — whether the disk is IDE or SCSI. Then under **IDE type** or **SCSI type**, click **Blank**. The Virtual Disk Configuration page appears.



4. In the **Disk File** field, enter the location for the virtual disk.
5. Specify the virtual device node in the **Virtual IDE Node** or **Virtual SCSI Node** list, as appropriate.
6. Decide if you want to make this virtual disk an independent disk. Under **Disk Mode**, check **Independent**, then check **Persistent** or **Nonpersistent**.

   Independent disks are not included in snapshots.

   Caution: The independent disk option should be used only by advanced users who need it for special-purpose configurations.

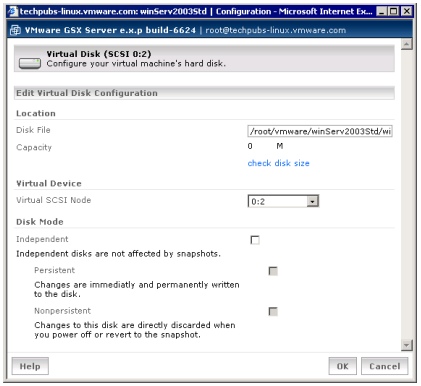
You have two options for an independent disk. You can make the disk **Persistent**, which means that changes are immediately and permanently written to the disk. Or you can make the disk **Nonpersistent**, which means that changes to the disk are discarded when you power off or reset the virtual machine.

7. Click **OK** to add the virtual disk.

Configuring a Virtual Disk from the Management Interface

To configure the virtual machine's virtual disk, complete the following steps.

1. In the Hardware page, under **Virtual Disk**, click **Edit**. The Virtual Disk page appears.



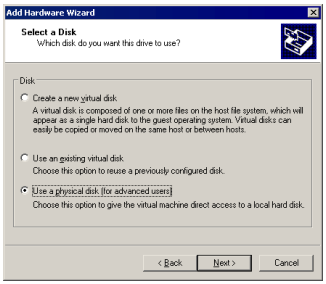
2. Change any of the following:
   - Change the virtual device node in the **Virtual IDE Node** or **Virtual SCSI Node** list, as appropriate.
   - Decide whether you want to make this virtual disk an independent disk. Under **Disk Mode**, check **Independent**, then check **Persistent** or **Nonpersistent**.
3. Click **OK** to save your changes and close the window.

Adding Physical Disks to a Virtual Machine

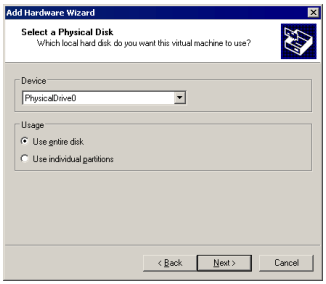
Use the virtual machine settings editor (**VM** > **Settings**) to add a physical disk to your virtual machine. The virtual machine should be powered off before you begin. If it is not, shut down the guest operating system normally, then click **Power Off** on the VMware Virtual Machine Console toolbar.

Caution: Physical disks are an advanced feature and should be configured only by advanced users.

1. Open the virtual machine settings editor (**VM** > **Settings**) and click **Add**. The Add Hardware Wizard guides you through the steps to create your virtual disk.
2. Click **Hard Disk**, then click **Next**. The Select a Disk screen appears.

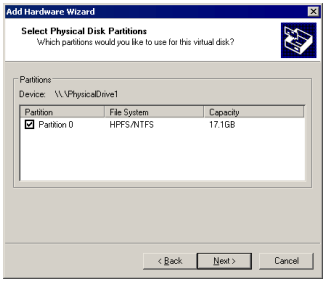

Add Hardware Wizard
Select a Disk
Which disk do you want this drive to use?
Disk
Create a new virtual disk
A virtual disk is composed of one or more files on the host file system, which will appear as a single hard disk to the guest operating system. Virtual disks can easily be copied or moved on the same host or between hosts.
Use an existing virtual disk
Choose this option to reuse a previously configured disk.
Use a physical disk (for advanced users)
Choose this option to give the virtual machine direct access to a local hard disk.
< Back
Next >
Cancel

3. Select **Use a physical disk**, then click **Next**. The Select a Physical Disk screen appears.



4. Choose the physical hard disk to use from the drop-down list. Then select whether you want to use the entire disk or use only individual partitions on the disk. Do one of the following:
   - To use the entire disk, select **Use entire disk**, then click **Next**.

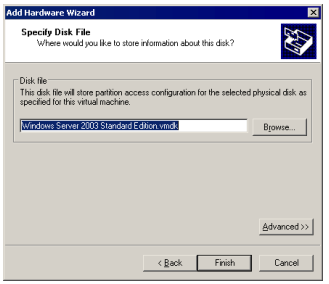
- To use specific partitions on the disk, select **Use individual partitions**, then click **Next**. The Select Partition screen appears.



Select which partitions you want to use in the virtual machine. Only the partitions you select in this step are visible to the virtual machine. All other partitions are hidden from it.

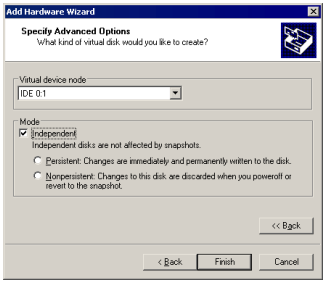
After you select the partitions, click **Next**.

5. The Specify Disk File screen appears.


Add Hardware Wizard
Specify Disk File
Where would you like to store information about this disk?
Disk file
This disk file will store partition access configuration for the selected physical disk as specified for this virtual machine.
Windows Server 2003 Standard Edition.vmdk
Browse...
Advanced >>
< Back
Finish
Cancel

Accept the default filename and location for the file that stores access information for this physical disk — or change it, if you want to use a different name or location. To find a different directory, click **Browse**.

Click **Advanced** if you want to specify the virtual machine SCSI or IDE device node to which this disk is connected.


Add Hardware Wizard
Specify Advanced Options
What kind of virtual disk would you like to create?
Virtual device node
IDE 0:1
Mode
Independent
Independent disks are not affected by snapshots.
Persistent: Changes are immediately and permanently written to the disk.
Nonpersistent: Changes to this disk are discarded when you poweroff or revert to the snapshot.
<< Back
< Back
Finish
Cancel

On the advanced settings screen, you can also specify a disk mode. This is useful in certain special-purpose configurations in which you want to exclude disks from the snapshot. For more information on the snapshot feature, see Taking Snapshots on page 153. You can choose between a normal disk and an independent disk.

Normal disks are included in snapshots. This is the default setting for a new disk.

Independent disks are not included in snapshots. If you select **Independent**, you must further select one of the following modes:

- **Persistent** — changes are immediately and permanently written to the disk.
- **Nonpersistent** — changes to the disk are discarded when you power off or revert to the snapshot.

When you have set the filename and location you want to use and have made any selections you want to make on the advanced settings screen, click **Finish**.

6. The wizard configures the new physical disk. If the partitions used on the physical disk are not formatted for your guest operating system, use the guest operating system's tools to format them.

Note: After you create a physical disk using one or more partitions on a physical disk, you should never modify the partition tables by running `fdisk` or a similar utility in the guest operating system.

Note: If you use `fdisk` or a similar utility on the host operating system to modify the partition table of the physical disk, you must recreate the virtual machine's physical disk.

Adding DVD-ROM or CD-ROM Drives to a Virtual Machine

You can add one or more DVD-ROM or CD-ROM drives to your virtual machine. You can connect the virtual machine's drive to a physical drive on the host machine or to an ISO image file.

You can configure the virtual DVD-ROM or CD-ROM drive as either IDE or SCSI, no matter what kind of physical drive you connect it to. In other words, if your host computer has an IDE CD-ROM drive, you can set up the virtual machine's drive as either SCSI or IDE and connect it to the host's drive. The same is true if the host's physical drive is a SCSI drive.

To add a new virtual DVD-ROM or CD-ROM drive to a virtual machine, make sure the virtual machine is powered off, then complete the following steps. You can add the device from the console or from the management interface.

Adding a DVD-ROM or CD-ROM Drive from the Console

1. Open the virtual machine settings editor (**VM** > **Settings**) and click **Add** to start the Add Hardware Wizard.
2. Click **DVD/CD-ROM Drive**, then click **Next**.
3. Select **Use physical drive** if you want to connect the virtual machine's drive to a physical drive on the host computer. Select **Use ISO Image** if you want to connect the virtual machine's drive to an ISO image file.
4. Do one of the following:
   - If you selected **Use physical drive**, choose the drive you want to use from the drop-down list or choose **Auto detect**.

     If you do not want the CD-ROM drive connected when the virtual machine starts, deselect **Connect at power on**.

     Click **Advanced** if you want to specify the device node the drive should use in the virtual machine.

     On the advanced settings screen you may also select **Legacy emulation**. This is necessary only if you have had problems using normal mode. The legacy emulation mode does not support all the capabilities of normal mode. For example, if you are using legacy emulation mode, you cannot record CDs, you cannot read multisession CDs, you cannot extract digital audio from a CD and you cannot read or write DVDs. For details, see Using Legacy Emulation for DVD-ROM and CD-ROM Drives on page 172.

     After you have made any desired changes in these settings, click **Finish**.

- If you selected **Use ISO Image**, enter the path and filename for the image file or click **Browse** to navigate to the file.

  If you do not want the CD-ROM drive connected when the virtual machine starts, deselect **Connect at power on**.

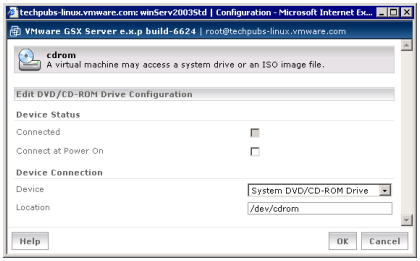
  Click **Advanced** if you want to specify the device node the drive should use in the virtual machine.

  After you have made any desired changes in these settings, click **Finish**.

5. The drive is set up initially so it appears to the guest operating system as an IDE drive. If you want it to appear to the guest operating system as a SCSI drive, click the drive's entry in the virtual machine settings editor and make the change.

Adding a DVD-ROM or CD-ROM Drive from the Management Interface

1. On the Hardware page, click **Add Device**. The Add Device Wizard starts.
2. Click **DVD/CD-ROM**. The cdrom page appears.



3. To connect this virtual machine to the host's DVD-ROM or CD-ROM drive when the virtual machine is powered on, check **Connect at Power On**.
4. Specify whether to connect to the host's DVD-ROM or CD-ROM drive or to an ISO image. In the **Device** list, select **System DVD/CD-ROM Drive** or **ISO Image**.
5. Enter the location of the drive or ISO image in the **Location** field. For example, the host's CD-ROM drive could be `D:` or `/dev/cdrom`.
6. Click **OK** to add the drive.

Adding Floppy Drives to a Virtual Machine

You can add floppy drives to your virtual machine, up to a total of two floppy drives. A virtual floppy drive can connect to a physical floppy drive on the host computer, to an existing floppy image file or to a blank floppy image file.

Adding a Floppy Drive from the Console

1. Open the virtual machine settings editor (**VM** > **Settings**) and click **Add** to start the Add Hardware Wizard.
2. Click **Floppy Drive**, then click **Next**.
3. Select what you want to connect to — a physical floppy drive on the host computer, an existing floppy image file or a new floppy image file. Click **Next**.
4. If you selected **Use a physical floppy drive**, choose the drive's letter (on a Windows host) or device name (on a Linux host) from the drop-down list, then click **Finish**.

   If you selected **Use a floppy image**, type the path and filename for the floppy image file you want to use, or click **Browse** to navigate to the file. Click **Finish**.

   If you selected **Create a blank floppy image**, use the default path and filename or type in a new one. To navigate to a location, click **Browse**. When the field contains the path and filename you want to use for the new floppy image file, click **Finish**.

Note: By default, only one floppy drive is enabled in the virtual machine's BIOS. If you are adding a second floppy drive to the virtual machine, click inside the virtual machine window and press F2 as the virtual machine boots to enter the BIOS setup utility. On the main screen, choose **Legacy Diskette B:** and use the plus (+) and minus (-) keys on the numerical keypad to select the type of floppy drive you want to use. Then press F10 to save your changes and close the BIOS setup utility.

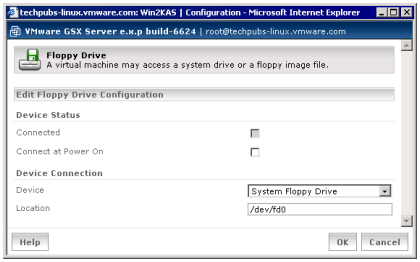
Adding a Floppy Drive from the Management Interface

If your server contains a floppy drive, you can add a virtual floppy drive to the virtual machine. You can point the floppy drive to a floppy disk image file.

A device can be connected to only one virtual machine on a server at a time.

To add a new virtual floppy drive to a virtual machine, make sure the virtual machine is powered off, then complete the following steps.

1. On the Hardware page, click **Add Device**. The Add Device Wizard starts.
2. Click **Floppy Drive**. The Floppy Drive page appears.



3. To have the floppy drive be connected to the virtual machine when you power it on, check **Connect at Power On**.
4. Specify whether to connect to the server's floppy drive or to a floppy image. In the **Device** list, select **System Floppy Drive** or **Floppy Image**.
5. Enter the location of the drive or floppy image in the **Location** field. For example, the server's floppy drive could be `A:` or `/dev/fd0`.
6. Click **OK** to add the drive.

Using VMware Virtual Disk Manager

VMware Virtual Disk Manager is a utility in GSX Server that allows you to create, manage and modify virtual disk files from the command line or within scripts.

One key feature is the ability to enlarge a virtual disk so its maximum capacity is larger than it was when you created it. This way, if you find you need more disk space on a given virtual disk, but do not want to add another virtual disk or use ghosting software to transfer the data on a virtual disk to a larger virtual disk, you can simply change the maximum size of the disk. This is something you cannot do with physical hard drives.

Another feature allows you to change whether or not all virtual disk space is preallocated or growable, and whether or not the virtual disk is stored in a single file or split into 2GB files. For example, you might find that you preallocated all the disk space for a virtual disk, but need to reclaim some hard disk space on the host. You can convert the preallocated virtual disk into a growable disk and remove the original virtual disk file. The new virtual disk is large enough to contain all the data on the original virtual disk. The virtual disk grows in size as you add data to it, as if you never preallocated the disk space when you created the virtual disk.

These features and the ability to use scripting to automate management of virtual disks were not possible with VMware products until now.

You can use the virtual disk manager to:

- Automate the management of virtual disks with scripts.
- Create virtual disks that are not associated with a particular virtual machine, to be used for templates, for example.
- Switch the virtual disk type from preallocated to growable, or vice versa. When changing the disk type to growable, some space on the virtual disk is reclaimed. You can shrink the virtual disk to reclaim even more disk space.
- Expand the size of a virtual disk so it is larger than the size specified when you created it.
- Defragment virtual disks.
- Prepare and shrink virtual disks without powering on the virtual machine (Windows hosts only).
- Rename and move virtual disks.

You can use the virtual disk manager with virtual disks created under VMware GSX Server, VMware Workstation and VMware VirtualCenter (provided the virtual disk was created on a GSX Server host managed by VirtualCenter).

You cannot use the virtual disk manager to create physical (raw) disks. Physical disks cannot be shrunk, neither by the virtual disk manager nor by GSX Server.

The following sections provide more information about the virtual disk manager:

- Running the VMware Virtual Disk Manager Utility on page 192
- Shrinking Virtual Disks with VMware Virtual Disk Manager on page 195
- Examples Using the VMware Virtual Disk Manager on page 195

Running the VMware Virtual Disk Manager Utility

To run the VMware Virtual Disk Manager utility, open a command prompt or terminal on the GSX Server host. For Windows hosts, change to the directory where you installed your GSX Server software. By default, this directory is `C:\Program Files\VMware\VMware GSX Server`.

The command syntax is:

```
vmware-vdiskmanager [options]
```

The options you can or must use include:

Options/Parameters	Description
`<diskname>`	Is the name of the virtual disk file. The virtual disk file must have a `.vmdk` extension. You can specify a path to where you want to locate the disk. If you mapped network shares on your host, you can create the virtual disk there by providing the correct path information with the disk filename.
`-c`	Creates the virtual disk. You must use the `-a`, `-s` and `-t` options, and you must specify the name of the virtual disk (`<diskname>`).

Options/Parameters	Description
`-r <sourcediskname> <targetdiskname>`	Converts the virtual disk specified by`<sourcediskname>`, creating a new virtual disk as a result. You must use the `-t` option to specify the disk type to which the virtual disk is converted and you must specify the name of the target virtual disk (`<targetdiskname>`). Once the conversion is completed and you have tested the converted virtual disk to make sure it works as expected, you can delete the original virtual disk file. In order for the virtual machine to recognize the converted virtual disk, you should use the virtual machine settings editor to remove the existing virtual disk from the virtual machine, then add the converted disk to the virtual machine. For information on adding virtual disks to a virtual machine, see Adding Virtual Disks to a Virtual Machine on page 177.
`-x <n>[GB|MB] <diskname>`	Expands the virtual disk to the specified capacity. You must specify the new, larger size of the virtual disk in Gigabytes or Megabytes. You cannot change the size of a physical (raw) disk. **Caution:** Before running the virtual disk manager utility, you should back up your virtual disk files. **Note:** If the virtual disk is partitioned, you must use a third-party utility in the virtual machine to expand the size of the partitions. For more information, see VMware knowledge base article 1647 at *www.vmware.com/support/kb/enduser/std_adp.php?p_faqid=1647*. If you have a virtual machine with a snapshot or a redo log stored in a different directory, do not use the virtual disk manager to expand the virtual disk until you remove the snapshot or commit the redo log. Otherwise, you may not be able to power on the virtual machine.
`-n <sourcediskname> <targetdiskname>`	Renames the virtual disk specified by `<sourcediskname>`. You must specify the name of the target virtual disk (`<targetdiskname>`). By providing directory paths, you can rename the disk and place it in a different directory or place the disk with the same name in a different directory. Before you rename the virtual disk or change the directory in which it is located, you should remove the virtual disk from any virtual machine that contains the disk. Choose **VM** > **Settings** > **<virtualdisk>**, then click **Remove**. If this virtual machine has a snapshot or a redo log stored in a different directory, remove the snapshot or commit the redo log. Otherwise, you may not be able to power on the virtual machine. After you rename or relocate the virtual disk, add it back to any virtual machines that use it. Choose **VM** > **Settings**, click **Add**, then follow the wizard to add this existing virtual disk.

Options/Parameters	Description
`-d <diskname>`	Defragments the specified virtual disk. You can defragment only growable virtual disks. You cannot defragment preallocated virtual disks.
`-p <mountpoint>`	Prepares a virtual disk for shrinking. If the virtual disk is partitioned into volumes, each volume must be prepared separately. The volume must be mounted by VMware DiskMount at `<mountpoint>`. After you prepare the volume, unmount it with VMware DiskMount. Continue mounting each volume of the virtual disk and preparing it for shrinking until you complete this process for all the volumes of the virtual disk. You can mount only one volume of a virtual disk at a time with VMware DiskMount. You can prepare volumes of virtual disks for shrinking on Windows hosts only.
`-k <diskname>`	Shrinks the specified virtual disk. You can shrink only growable virtual disks. You can shrink virtual disks on Windows hosts only. You cannot shrink a virtual disk if the virtual machine has a snapshot. To keep the virtual disk in its current state, simply remove the snapshot. To discard changes made since you took the snapshot, revert to the snapshot.
`-a [ide|buslogic|lsilogic]`	Specifies the disk adapter type. You must specify an adapter type when creating a new virtual disk. Choose one of the following types: • `ide` — for an IDE adapter. • `buslogic` — for a BusLogic SCSI adapter. • `lsilogic` — for an LSI Logic SCSI adapter.
`-s <n>[GB|MB]`	Specifies the size of the virtual disk. Specify whether the size `<n>` is in `GB` (Gigabytes) or `MB` (Megabytes). You must specify the size of a virtual disk when you create it. Even though you must specify the size of a virtual disk when you expand it, you do not use the `-s` option at that time.
`-t [0|1|2|3]`	You must specify the type of virtual disk when you create a new one or reconfigure an existing one. Specify one of the following disk types: `0` — to create a single, growable virtual disk. `1` — to create a growable virtual disk split into 2GB files. `2` — to create a preallocated virtual disk contained in a single virtual disk file. `3` — to create a preallocated virtual disk split into 2GB files.
`-q`	Disables virtual disk manager logging. If you keep logging enabled, messages generated by the virtual disk manager are stored in a log file. The name and location of the log file appear in the terminal after the virtual disk manager command is run.

Shrinking Virtual Disks with VMware Virtual Disk Manager

If the virtual disk is located on a Windows host, you can use the virtual disk manager to prepare and shrink virtual disks. You cannot use the virtual disk manager to prepare or shrink virtual disks located on a Linux host. You cannot use the virtual disk manager to shrink physical disks. Shrinking a virtual disk does not reduce the maximum capacity of the virtual disk itself. For more information about shrinking, see Defragmenting and Shrinking Virtual Disks on page 168.

Caution: You cannot shrink a virtual disk if the virtual machine has a snapshot. To keep the virtual disk in its current state, simply remove the snapshot. To discard changes made since you took the snapshot, revert to the snapshot.

You must prepare each volume of the virtual disk for shrinking before you can shrink the disk. To prepare a volume for shrinking, you must first mount it. To mount the volume, use the VMware DiskMount Utility, available as a free download from the VMware Web site. Go to *www.vmware.com/download/diskmount.html*.

The VMware DiskMount user's manual is available from the VMware Web site at *www.vmware.com/pdf/VMwareDiskMount.pdf*. It contains instructions on mounting and unmounting virtual disk volumes with DiskMount.

VMware DiskMount mounts individual volumes of a virtual disk. For optimal shrinking of a virtual disk, you should mount all the volumes and prepare them for shrinking.

After you mount a virtual disk volume, use the virtual disk manager to prepare the volume for shrinking. Once you prepare a volume, unmount it, then repeat the process for each volume of the virtual disk. After you prepare all the volumes of the virtual disk, you can shrink the virtual disk. For examples, see Preparing a Virtual Disk for Shrinking on page 197 and Shrinking a Virtual Disk on page 197.

Examples Using the VMware Virtual Disk Manager

The following examples illustrate how to use the virtual disk manager. You run the virtual disk manager from a command prompt.

Creating a Virtual Disk

To create a new virtual disk, use the following:

```
vmware-vdiskmanager -c -t 0 -s 40GB -a ide myDisk.vmdk
```

This creates a 40GB IDE virtual disk named `myDisk.vmdk`. The virtual disk is contained in a single `.vmdk` file. The disk space is not preallocated.

Converting a Virtual Disk

To convert a virtual disk from preallocated to a growable disk, use the following:

```
vmware-vdiskmanager -r sourceDisk.vmdk -t 0 targetDisk.vmdk
```

This converts the disk from its original preallocated type to a growable virtual disk consisting of a single virtual disk file. All of the virtual disk space is no longer preallocated, and the virtual disk manager reclaims some disk space in the virtual disk so it is only as large as the data contained within it.

Expand the Size of an Existing Virtual Disk

To expand the size of a virtual disk, use the following:

```
vmware-vdiskmanager -x 40GB myDisk.vmdk
```

This increases the maximum capacity of the virtual disk to 40GB.

Renaming a Virtual Disk

To rename a virtual disk, first remove it from any virtual machine that contains the disk (choose **VM** > **Settings** > **<virtualdisk>**, then click **Remove**).

Then use the following:

```
vmware-vdiskmanager -n myDisk.vmdk myNewDisk.vmdk
```

To rename the disk and locate it in a different directory, use:

```
vmware-vdiskmanager -n myDisk.vmdk
..\<new>\<path>\myNewDisk.vmdk
```

Note: The paths used in these examples assume a Windows host.

To locate the disk in a different directory but keep the same name, use:

```
vmware-vdiskmanager -n myDisk.vmdk
..\<new>\<path>\myDisk.vmdk
```

After you rename or relocate the virtual disk, add it back to any virtual machines that use it. Choose **VM** > **Settings**, click **Add**, then follow the wizard to add this existing virtual disk.

Defragmenting a Virtual Disk

To defragment a virtual disk, use the following:

```
vmware-vdiskmanager -d myDisk.vmdk
```

Remember, you cannot defragment a virtual disk if you preallocated all the disk space when you created the virtual disk. You cannot defragment a physical disk.

Preparing a Virtual Disk for Shrinking

Before you can shrink a virtual disk, you must prepare each volume on the disk for shrinking. To prepare a volume, it must be located on a Windows host. First you must mount the volume. To mount the volume, use the VMware DiskMount Utility, available as a free download from the VMware Web site. For information about downloading and using VMware DiskMount, see Shrinking Virtual Disks with VMware Virtual Disk Manager on page 195.

VMware DiskMount mounts individual volumes of a virtual disk. For optimal shrinking of a virtual disk, you should mount all the volumes and shrink them.

After you mount a virtual disk volume, use the virtual disk manager to prepare the disk for shrinking. To prepare the volume mounted at the M drive for shrinking, use the following:

```
vmware-vdiskmanager -p M:
```

Once the preparations are complete, unmount the volume. Repeat this process for each volume of the virtual disk. After you prepare all the volumes for shrinking, you can shrink the virtual disk.

Shrinking a Virtual Disk

To shrink a virtual disk, it must be located on a Windows host. Before you can shrink the virtual disk, make sure you prepare all the volumes of the virtual disk for shrinking. Then use the following:

```
vmware-vdiskmanager -k myDisk.vmdk
```

Remember, you cannot shrink a virtual disk if you preallocated all the disk space when you created the virtual disk. You cannot shrink a physical (raw) disk.

You cannot shrink a virtual disk if the virtual machine has a snapshot. To keep the virtual disk in its current state, simply remove the snapshot. To discard changes made since you took the snapshot, revert to the snapshot.

Configuring a Dual-Boot Computer for Use with a Virtual Machine

Many users install GSX Server on a dual-boot or multiple-boot computer so they can run one or more of the existing operating systems in a virtual machine. If you are doing this, you may want to use the existing installation of an operating system rather than reinstall it in a virtual machine.

To support such installations, GSX Server makes it possible for you to use a physical IDE disk or partition, also known as a raw disk, inside a virtual machine.

Note: GSX Server supports booting from physical disk partitions only on IDE drives. Booting guest operating systems from raw SCSI drives is not supported. For a discussion of the issues on a Linux host, see Configuring Dual- or Multiple-Boot SCSI Systems to Run with VMware GSX Server on a Linux Host on page 214.

Caution: You cannot use a physical disk that is stored on a SAN. You must use a disk or a partition on the GSX Server host.

Setting up a physical disk configuration for a virtual machine is more complicated than using a virtual disk. Virtual disks are recommended unless you have a specific need to run directly from a physical disk or partition.

Caution: Physical disks are an advanced feature and should be configured only by advanced users.

Using the Same Operating System in a Virtual Machine and on the Host Computer

You may sometimes want to run an operating system inside a virtual machine and at other times want to run that same installation of the operating system by booting the host computer directly into that operating system. If you want to use this approach, you must be aware of some special considerations

The issues arise because the virtual hardware that the operating system sees when it is running in a virtual machine is different from the physical hardware it sees when it is running directly on the host computer. It is as if you were removing the boot drive from one physical computer and running the operating system installed there in a second computer with a different motherboard, video card and other peripherals — then moving it back and forth between the two systems.

The general approach for resolving these issues is to set up profiles for each of the two operating environments — the virtual machine and the physical computer. You can then choose the appropriate profile when you start the operating system. On some

hardware, however, booting a previously installed operating system within a virtual machine may not work.

Technical notes in this section document the issues most commonly encountered with various guest operating systems. Read the notes that apply to your guest operating system before you begin to set up your virtual machine.

Before You Begin

Before you begin, be sure to read all the sections listed under the name of the operating system you intend to run as a guest in a virtual machine.

Windows Server 2003

Caution: Running a Windows Server 2003 guest from a physical disk is not supported. You should not test a Windows Server 2003 physical disk configuration in a production environment.

- Configuring Dual- or Multiple-Boot Systems to Run with GSX Server on page 200
- Setting Up Hardware Profiles in Virtual Machines on page 206
- Running a Windows 2000, Windows XP or Windows Server 2003 Virtual Machine from an Existing Multiple-Boot Installation on page 210
- Do Not Use Windows 2000, Windows XP and Windows Server 2003 Dynamic Disks as Physical Disks on page 213

Windows XP

Caution: Running a Windows XP guest from a physical disk is not supported. You should not test a Windows XP physical disk configuration in a production environment.

- Configuring Dual- or Multiple-Boot Systems to Run with GSX Server on page 200
- Setting Up Hardware Profiles in Virtual Machines on page 206
- Running a Windows 2000, Windows XP or Windows Server 2003 Virtual Machine from an Existing Multiple-Boot Installation on page 210
- Do Not Use Windows 2000, Windows XP and Windows Server 2003 Dynamic Disks as Physical Disks on page 213

Windows 2000

- Configuring Dual- or Multiple-Boot Systems to Run with GSX Server on page 200
- Setting Up Hardware Profiles in Virtual Machines on page 206
- Running a Windows 2000, Windows XP or Windows Server 2003 Virtual Machine from an Existing Multiple-Boot Installation on page 210

- Do Not Use Windows 2000, Windows XP and Windows Server 2003 Dynamic Disks as Physical Disks on page 213

Windows NT

- Configuring Dual- or Multiple-Boot Systems to Run with GSX Server on page 200
- Setting Up Hardware Profiles in Virtual Machines on page 206

Windows 98

- Configuring Dual- or Multiple-Boot Systems to Run with GSX Server on page 200
- Setting Up Hardware Profiles in Virtual Machines on page 206
- Setting Up the SVGA Video Driver for Use with a Windows 98 Guest Operating System Booted from a Physical Disk on page 211

Windows 95

- Configuring Dual- or Multiple-Boot Systems to Run with GSX Server on page 200
- Setting Up Hardware Profiles in Virtual Machines on page 206
- Setting Up the SVGA Video Driver for a Windows 95 Guest Operating System Booted from a Physical Disk on page 210

SCSI Systems Using a Linux Host

- Configuring Dual- or Multiple-Boot SCSI Systems to Run with VMware GSX Server on a Linux Host on page 214

Other Uses of Physical Disks

It is also possible to install a guest operating system on a physical disk when you plan to use that disk only within a virtual machine. For details on setting up a such a configuration, see Installing an Operating System onto a Physical Partition from a Virtual Machine on page 220.

Configuring Dual- or Multiple-Boot Systems to Run with GSX Server

GSX Server uses description files to control access to each raw IDE device on the system. These description files contain access privilege information that controls a virtual machine's access to certain partitions on the disks. This mechanism prevents users from accidentally running the host operating system again as a guest or running a guest operating system that the virtual machine was not configured to use. The description file also prevents accidental corruption of physical disk partitions by badly behaved operating systems or applications.

Use the New Virtual Machine Wizard or the VMware Management Interface to configure GSX Server to use existing physical disk partitions. The wizard guides you though creating a configuration for a new virtual machine including configuring the physical disk description files. Typically, you rerun the wizard to create a separate configuration for each guest operating system installed on a raw partition.

If a boot manager is installed on the computer system, the boot manager runs inside the virtual machine and presents you with the choice of guest operating systems to run. You must manually choose the guest operating system that this configuration was intended to run.

Running Windows Guests on Windows Hosts with FAT File Systems

There is a potential problem with GSX Server on Windows hosts when you boot an operating system from an existing partition. If the Windows host's partition uses a FAT file system, the guest operating system (for example, Windows 98 or Windows 95) sees this partition at boot time and attempts to fix the file system on that partition. This causes serious problems, because the host operating system is actively using that partition.

If you use an advanced boot manager such as BootMagic (PowerQuest) or System Commander (V Communications), it solves this problem by changing the partition type to "unknown." If you are already using such an advanced boot manager to dual boot, the boot manager's partition marking scheme works fine with GSX Server.

However, if you are not using an advanced boot manager for dual booting, the configuration process described below hides partitions that do not belong to the guest operating system. When physical disk partition hiding is enabled, all read-only partitions are mapped to "unknown." Also, all updates to the master boot record are intercepted and not written to the actual master boot record.

Windows 2000, Windows XP and Windows Server 2003 Dynamic Disks

If your host is running Windows 2000, Windows XP or Windows Server 2003 and is using dynamic disks, see Do Not Use Windows 2000, Windows XP and Windows Server 2003 Dynamic Disks as Physical Disks on page 213.

Using the LILO Boot Loader

If you are using the LILO boot loader and try to boot a virtual machine from an existing raw partition, you may see `L 01 01 01 01 01 01 ...` instead of a `LILO:` prompt. This can happen regardless of the host operating system. As part of booting a physical PC or a virtual machine, the BIOS passes control to code located in the master boot record (MBR) of the boot device. LILO begins running from the MBR, and in order to finish running correctly, it needs access to the native Linux partition where the rest

of LILO is located — usually the partition with the `/boot` directory. If LILO can't access the rest of itself, an error message like the one above appears.

To avoid the problem, follow the configuration steps below and be sure to select the native Linux partition where the rest of LILO is located. The next time the virtual machine tries to boot, the LILO code in the MBR should be able to access the rest of LILO and display the normal `LILO:` prompt.

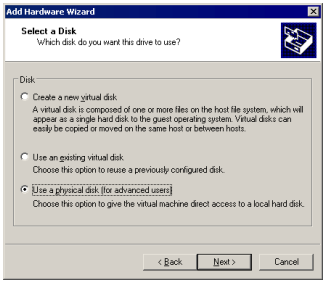
Configuring a Windows Host

Use the following steps to run a guest operating system from a physical disk.

Note: If you use a Windows host's IDE disk in a physical disk configuration, you must not configure it as the slave on the secondary IDE channel if the master on that channel is a CD-ROM drive.

1. If you are running a Windows guest operating system, read Setting Up Hardware Profiles in Virtual Machines on page 206. You should boot the guest operating system natively on the computer and create a hardware profile for the virtual machine before proceeding.
2. Create a separate configuration for each guest operating system.

   To configure a virtual machine to run from a physical disk or one of its partitions, start the New Virtual Machine Wizard (**File** > **New Virtual Machine**) and select **Custom**.
3. When you reach the Select a Disk step, select **Use a physical disk**.


Add Hardware Wizard
Select a Disk
Which disk do you want this drive to use?
Disk
Create a new virtual disk
A virtual disk is composed of one or more files on the host file system, which will appear as a single hard disk to the guest operating system. Virtual disks can easily be copied or moved on the same host or between hosts.
Use an existing virtual disk
Choose this option to reuse a previously configured disk.
Use a physical disk (for advanced users)
Choose this option to give the virtual machine direct access to a local hard disk.
< Back
Next >
Cancel

4. Complete the wizard steps, specifying the appropriate disk or partition to use for this virtual machine.

   Note: The maximum size of an IDE disk in a virtual machine is 128GB.
5. To run multiple guest operating systems from different physical disk partitions, unmap these partitions on the host.

Use Disk Management (**Start** > **Settings** > **Control Panel** > **Administrative Tools** > **Computer Management** > **Storage** > **Disk Management**). Select the partition you want to unmap, then from the **Action** menu select **All Tasks** > **Change Drive Letter and Path**. Click the **Remove** button.

6. Use the virtual machine settings editor (**VM** > **Settings**) if you want to change any configuration options from the wizard defaults — for example, to change the amount of memory allocated to the guest operating system.
7. If you have multiple IDE drives configured on a system, the VMware BIOS normally attempts to boot them in this sequence:

   a. Primary master

   b. Primary slave

   c. Secondary master

   d. Secondary slave

   If you have multiple SCSI drives configured on a system, the VMware BIOS normally attempts to boot them in the order of the SCSI device number.

   If you have both SCSI and IDE drives configured, the VMware BIOS normally attempts to boot SCSI drives followed by IDE drives, in the order listed above.

   The boot sequence can be changed in the **Boot** menu of the virtual machine's Phoenix BIOS. After powering on the virtual machine, press F2 during the BIOS boot in the virtual machine to enter the BIOS setup menu.
8. Power on the virtual machine. Click the **Power On** button. The virtual machine starts, runs the Phoenix BIOS, then boots from the master boot record (MBR).

   Choose the target operating system from the list of options offered by the boot manager.
9. Remember that your virtual machine hardware environment, which the guest operating system is about to run in for the first time, probably differs significantly from the physical hardware of your host computer.

   For Windows guest operating systems, Plug and Play reconfigures Windows. Set up your virtual hardware profile with the devices found and configured by Plug and Play. See Setting Up Hardware Profiles in Virtual Machines on page 206 for more information.
10. Install VMware Tools in your guest operating system.

Caution: If you take a snapshot while using your physical disk, you must either revert to the snapshot or remove the snapshot before you reboot your guest operating

system natively. This is necessary because any changes to sectors on the physical disk that have been modified on the disk invalidate the snapshot for the disk.

Configuring a Linux Host

1. If you are running a Windows guest operating system, read Setting Up Hardware Profiles in Virtual Machines on page 206. You should boot the guest operating system natively on the computer and create a hardware profile for the virtual machine before proceeding.
2. Create a separate configuration for each guest operating system.
3. Check operating system partition mounts. Be sure the existing disk partitions that you plan to configure the virtual machine to use are not mounted by Linux.
4. Set the device group membership or device ownership.

   The master physical disk device or devices need to be readable and writable by the user who runs GSX Server. On most distributions, the raw devices, such as `/dev/hda` (IDE physical disk) and `/dev/sda` (SCSI physical disk) belong to group ID `disk`. If this is the case, you can add GSX Server users to the `disk` group. Another option is to change the owner of the device. Please think carefully about security issues when exploring different options here.

   Often, the most convenient approach is to grant GSX Server users access to all `/dev/hd[abcd]` raw devices that contain operating systems or boot managers and then rely on GSX Server's physical disk configuration files to guard access. This provides boot managers access to configuration files and other files they may need to boot the operating systems. For example, LILO needs to read `/boot` on a Linux partition to boot a non-Linux operating system that may be on another drive. As noted above, you should consider the security implications of the configuration you choose.
5. If you plan to run a second Linux installation from an existing partition as a guest operating system and your physical computer's `/etc/lilo.conf` has a memory register statement such as `Append= "mem…"`, you may want to adjust the append memory parameter or create a new entry in LILO for running Linux in a virtual machine.

   If the amount of memory configured in `lilo.conf` exceeds the amount of memory assigned to the virtual machine, the guest operating system is likely to panic when the virtual machine tries to boot the second Linux installation.

   You can create another entry in `lilo.conf` for running Linux in a virtual machine by specifying a different amount of memory than what would normally be recognized when Linux boots directly on the physical machine.

6. To configure a virtual machine to run from a physical disk partition, start the New Virtual Machine Wizard (**File** > **New** > **New Virtual Machine**) and select **Custom**.
7. When you reach the Select a Disk step, select **Use a physical disk**.
8. Complete the wizard steps, specifying the appropriate disk or partition to use for this virtual machine.

   Caution: Corruption is possible if you allow the virtual machine to modify a partition that is simultaneously mounted under Linux. Since the virtual machine and guest operating system access an existing partition while the host continues to run Linux, it is critical that the virtual machine not be allowed to modify any partition mounted under Linux or in use by another virtual machine.

   To safeguard against this problem, be sure the partition you use in the virtual machine is not mounted under the Linux host.
9. Complete the remaining steps in the wizard.
10. If you have multiple IDE drives configured on a system, the VMware BIOS normally attempts to boot them in this sequence:

    a. Primary master

    b. Primary slave

    c. Secondary master

    d. Secondary slave

    If you have multiple SCSI drives configured on a system, the VMware BIOS normally attempts to boot them in the order of the SCSI device number.

    If you have both SCSI and IDE drives configured, the VMware BIOS normally attempts to boot SCSI drives followed by IDE drives, in the order listed above.

    You can change the boot sequence using the **Boot** menu of the virtual machine's Phoenix BIOS. To enter the BIOS setup utility, power on the virtual machine and press F2 as the virtual machine begins to boot.
11. Power on the virtual machine. Click the **Power On** button. The virtual machine starts, runs the Phoenix BIOS, then boots from the master boot record (MBR).

    Choose the target operating system from the list of options offered by the boot manager.
12. Remember that your virtual machine hardware environment, which the guest operating system is about to run in for the first time, probably differs significantly from the physical hardware of your machine.

For Windows guest operating systems, Plug and Play reconfigures Windows. Set up your virtual hardware profile with the devices found and configured by Plug and Play. See Setting Up Hardware Profiles in Virtual Machines on page 206 for more information.

13. Install VMware Tools in your guest operating system.

Warning: If you take a snapshot while using your physical disk, you must either revert to the snapshot or remove the snapshot before you reboot your guest operating system natively. This is necessary because any changes to sectors on the physical disk that have been modified on the disk invalidate the snapshot for the disk.

Setting Up Hardware Profiles in Virtual Machines

Certain operating systems use hardware profiles to load the appropriate drivers for a given set of hardware devices. If you have a dual-boot system and want to use a virtual machine to boot a previously installed operating system from an existing partition, you must set up "physical" and "virtual" hardware profiles.

Only users who are familiar with GSX Server virtual machines and the Windows hardware profiles concept should attempt this.

If you haven't already done so, review Configuring Dual- or Multiple-Boot Systems to Run with GSX Server on page 200 before proceeding.

Each virtual machine provides a platform that consists of the following virtual devices:

- Virtual DVD/CD-ROM
- Virtual IDE and SCSI hard disk drives
- Standard PCI graphics adapter
- Standard floppy disk drive
- Intel 82371 PCI Bus Master IDE controller (includes primary and secondary IDE controllers)
- BusLogic BT-958 compatible SCSI host adapter
- Standard 101/102-key keyboard
- PS/2-compatible mouse
- AMD PCnet-PCI II compatible Ethernet adapter
- Serial ports (COM1-COM4)
- Parallel ports (LPT1-LPT2)
- Two-port USB hub
- Sound card compatible with the Sound Blaster AudioPCI

- 82093AA IOAPIC

This set of virtual devices is different from the set of physical hardware devices on the host computer and is independent of the underlying hardware with a few exceptions (the processor itself is such an exception). This feature provides a stable platform and allows operating system images installed within a virtual machine to be migrated to other physical machines, regardless of the configuration of the physical machine.

If an operating system is installed directly into a GSX Server virtual machine, the operating system properly detects all the virtual devices by scanning the hardware. However, if an operating system is already installed on the physical computer (for example, in a dual-boot configuration), the operating system already is configured to use the physical hardware devices. In order to boot such a preinstalled operating system in a virtual machine, you need to create separate hardware profiles in order to simplify the boot process.

Microsoft Windows operating systems, beginning with Windows 95 and Windows NT 4.0, allow you to create hardware profiles. Each hardware profile is associated with a set of known devices. If more than one hardware profile exists, the system prompts the user to choose between different hardware profiles at boot time.

Windows 95, Windows 98, Windows Me, Windows 2000, Windows XP and Windows Server 2003 use Plug and Play at boot time to confirm that the actual devices match the chosen hardware profile. Mismatches lead to the automatic detection of new devices. Although this operation succeeds, it can be fairly slow.

Windows NT does not have Plug and Play support and uses the hardware profiles to initialize its devices. Mismatches lead to errors reported by the device drivers and the devices are disabled.

In order to set up hardware profiles for your physical and virtual machines, follow these steps:

1. Before running GSX Server to boot an operating system previously installed on a disk partition, boot the operating system natively and create two hardware profiles, which you can call Physical Machine and Virtual Machine. To do this, open **Control Panel** > **System**, then click the **Hardware Profiles** tab — or click the **Hardware** tab, then click **Hardware Profiles**, depending on the operating system. Click the **Copy** button and name the copies appropriately.
2. **Windows NT only:** While still running the operating system natively, use the Device Manager to disable some devices from the Virtual Machine hardware profile. To do this, open **Control Panel** > **Devices**, then select the individual devices to disable. Devices to disable in the Virtual Machine hardware profile

include audio, MIDI and joystick devices, Ethernet and other network devices and USB devices. Remember to disable them in the Virtual Machine hardware profile only.

Skip this step if you are running Windows 95, Windows 98, Windows Me, Windows 2000, Windows XP or Windows Server 2003. The initial Plug and Play phase detects device mismatches.

3. Reboot the computer into your intended host operating system — for example, into Linux if you are running GSX Server on a Linux host.
4. Use the New Virtual Machine Wizard to configure your virtual machine as described in Configuring Dual- or Multiple-Boot Systems to Run with GSX Server on page 200.
5. Boot the virtual machine and use your existing boot manager to select the guest operating system. Choose Virtual Machine at the hardware profile menu prompt. You encounter device failure messages and delays during this initial boot.
6. **Windows Server 2003, Windows XP and Windows 2000 guests:** After you log on to Windows Server 2003, Windows XP or Windows 2000 (now running as a guest operating system) you should see a Found New Hardware dialog box for the video controller as Plug and Play runs and discovers the virtual hardware. Do not install drivers at this time. Click **Cancel** to close the Found New Hardware dialog box.

   Do not reboot the virtual machine. Click **No** in the System Settings Change/ Reboot dialog box.

   Windows Server 2003, Windows XP and Windows 2000 automatically detect and load the driver for the AMD PCnet PCI Ethernet card. At this point, you should install VMware Tools inside the virtual machine. Allow the virtual machine to reboot after VMware Tools has been installed. Once Windows Server 2003, Windows XP or Windows 2000 reboots inside the virtual machine, select a new SVGA resolution from the Settings tab of the Display Properties dialog box to increase the size of the virtual machine's display window.

   If you want to enable the virtual machine's sound adapter to work inside the guest operating system, finish the remaining steps in this section, then refer to Configuring Sound on page 337.

   Windows NT guests only: After the operating system has finished booting in the virtual machine, view the event log to see which physical devices have failed to start properly. You can disable them from the Virtual Hardware profile using the Device Manager (**Control Panel** > **Devices**).

If you want to enable the virtual machine's sound adapter to work inside the Windows NT guest operating system, finish the remaining steps in this section, then refer to Configuring Sound on page 337.

Windows 95 and Windows 98 guests: You should see New Hardware Detected dialog boxes as Plug and Play runs and discovers the virtual hardware. Windows prompts you for locations to search for device drivers. Most of the device drivers are available in the existing operating system installation, but you may need the installation CD-ROM for some networking device drivers. Windows also asks you to reboot your system several times as it installs the device drivers.

In some instances, Windows may not recognize the CD-ROM drive when it prompts you to insert the CD-ROM to look for device drivers during the initial hardware detection. In such cases, you can cancel the installation of the particular device or try pointing to `C:\Windows\system\` to search for device drivers on the hard disk. Any failed device installations may be performed at a later time after the CD-ROM drive is recognized.

After Windows has installed the virtual hardware and its drivers, you can remove the failed devices corresponding to the physical hardware using the Device Manager (**Control Panel** > **System** > **Device Manager**).

Select the device, then click the **Remove** button. If a device appears in multiple hardware profiles, you can select the hardware profile or profiles from which to remove the device.

If you want to enable the virtual machine's sound adapter to work inside the guest operating system, finish the remaining steps in this section, then refer to Configuring Sound on page 337.

7. Confirm that your virtual devices — specifically, the network adapter — are working properly.

   Windows 95 and Windows 98 guests: If any virtual device is missing, you can detect it by running **Control Panel** > **Add New Hardware**.

8. Install VMware Tools. VMware Tools appears and runs in both hardware configurations but affects only the virtual machine.

   Note: The next time you reboot Windows natively using the Physical Machine hardware profile, some virtual devices may appear in the device list. You can disable or remove these virtual devices from the Physical Machine hardware profile in the same way that you removed physical devices from the Virtual Machine hardware profile in step 6, above.

Running a Windows 2000, Windows XP or Windows Server 2003 Virtual Machine from an Existing Multiple-Boot Installation

If you have installed Windows 2000, Windows XP or Windows Server 2003 on a computer, then try to run that same installation of the operating system as a GSX Server virtual machine running from a physical disk, the virtual machine may fail with an error message reporting an inaccessible boot device.

The problem occurs because the physical computer and the virtual machine require different IDE drivers. The Windows plug and play feature, which handles drivers for many hardware devices, does not install new IDE drivers.

If you encounter this problem, VMware recommends that you install your Windows 2000, Windows XP or Windows Server 2003 guest operating system in a virtual disk, rather than running it from a physical disk.

If you encounter this problem but it is important for you to run the virtual machine from the existing physical disk configuration, you can set up separate hardware profiles (described in Setting Up Hardware Profiles in Virtual Machines on page 206) and manually update the IDE driver in the profile for the virtual machine. For a detailed description of the workaround, see the VMware knowledge base (*www.vmware.com/info?id=41*).

Setting Up the SVGA Video Driver for a Windows 95 Guest Operating System Booted from a Physical Disk

This section explains how to configure the video driver in a Windows 95 physical disk installation using GSX Server. The steps below assume you are using Windows 95 as one of the operating systems in a dual-boot or multiple-boot configuration. Following these steps, you create separate hardware profiles for your virtual machine and your physical machine. For more details on hardware profiles, see Setting Up Hardware Profiles in Virtual Machines on page 206.

1. Boot Windows 95 natively (not in a virtual machine).
2. Right-click the **My Computer** icon on the desktop, then select **Properties**.
3. Click the **Hardware Profiles** tab.
4. Highlight the **Original Configuration** profile, then click **Copy**.
5. Name the profile Virtual Machine, then click **OK**. You may also want to rename the Original Configuration profile to Physical Machine.
6. Click **OK** to close the System Properties dialog box.

7. Shut down Windows 95 and reboot the system.
8. Boot into your host operating system (Linux, Windows 2000 or Windows Server 2003).
9. Start the Windows 95 virtual machine.
10. Select **Virtual Machine** from the list of profiles when prompted.
11. If you are prompted to select the CPU Bridge, accept the default, then click **OK**.
12. Restart Windows 95 when prompted.
13. Again, select **Virtual Machine** from the list of profiles when prompted.
14. When the video card is detected, you are prompted to select which driver you want to install for your new hardware. Click the **Select from a list of alternate drivers** radio button, then click **OK**.
15. Select **Display Adapters** from the Select Hardware Type dialog box.
16. Select **Standard Display Adapter (VGA)** from the device list, then click **OK**.
17. Restart Windows 95 when prompted.
18. Install VMware Tools as outlined in Installing VMware Tools on page 62, then restart the virtual machine.
19. Start the Device Manager and expand the **Display adapters** tree.
20. Highlight **VMware SVGA**. Click **Properties**.
21. Clear the **Physical Machine** check box, then click **OK**. Click **Close**.
22. Shut down Windows 95 and power off the virtual machine.
23. Shut down your host operating system (Linux, Windows 2000 or Windows Server 2003) and reboot into Windows 95.
24. Select the **Physical Machine** profile when prompted.
25. Repeat steps 19 through 21 and uncheck **Virtual Machine**, leaving **Physical Machine** checked.

Setting Up the SVGA Video Driver for Use with a Windows 98 Guest Operating System Booted from a Physical Disk

This section explains how to configure the video driver in a Windows 98 physical disk installation using GSX Server. The steps below assume you are using Windows 98 as one of the operating systems in a dual-boot or multiple-boot configuration. Following these steps, you create separate hardware profiles for your virtual machine and your

physical machine. For more details on hardware profiles, see Setting Up Hardware Profiles in Virtual Machines on page 206.

1. Boot Windows 98 natively (not in a virtual machine).
2. Right-click the **My Computer** icon on the desktop, then select **Properties**.
3. Click the **Hardware Profiles** tab.
4. Highlight the **Original Configuration** profile, then click **Copy**.
5. Name the profile Virtual Machine, then click **OK**.

   You may also want to rename the Original Configuration profile to Physical Machine.
6. Click **OK** to close the System Properties dialog box.
7. Shut down Windows 98 and reboot the system.
8. Boot into your host operating system (Linux, Windows 2000 or Windows Server 2003).
9. Select **Virtual Machine** from the list of profiles when prompted.
10. Windows 98 auto-detects the virtual machine's devices and installs the device drivers.
11. When Windows detects the video card driver, select **Search for the best driver**.
12. When prompted to reboot, click **No**. The AMD PCNET driver is installed, followed by the IDE controller drivers.
13. When prompted to reboot, click **Yes**.
14. Select the **Virtual Machine** hardware profile.
15. After Windows 98 has completed booting, start the Add New Hardware Wizard from the Control Panel.
16. Click **Next**, then **Next** again.
17. Select **No, the device isn't in the list**.
18. Click **Yes**, then click **Next**.
19. After all devices have been detected, click the **Details** button to list the detected non-Plug and Play devices.
20. Click **Finish**, then reboot the virtual machine when prompted.
21. Select the **VMware GSX Server** configuration profile. Notice that an unknown monitor is detected and installed.
22. Install VMware Tools as outlined in Installing VMware Tools on page 62.

23. Open the Device Manager. It should show that you have
    - Standard PCI Graphics Adapter
    - VMware SVGA Display Adapter
24. Shut down the Windows 98 virtual machine and your host operating system.
25. Boot natively into Windows 98, then start the Device Manager.
26. Select the **VMware SVGA** device if listed, then click **Remove**.
27. Select the **Remove from Specific Configuration** radio button, then select **Physical Machine** from the configuration list.
28. Click **OK**, then reboot Windows 98 when prompted.
29. Boot into Windows 98 natively and verify the display settings. You should be able to use the display driver that you installed natively before starting this procedure.

Do Not Use Windows 2000, Windows XP and Windows Server 2003 Dynamic Disks as Physical Disks

Windows 2000, Windows XP and Windows Server 2003 support a disk type called a dynamic disk. Dynamic disks use a proprietary Microsoft format for recording partition information. This format is not publicly documented and thus is not supported for use in physical disk configurations under GSX Server.

Windows 2000, Windows XP and Windows Server 2003 also support the older type of partition table. Disks that use this type of partition table are called basic disks.

You can use the disk management tool to check the type of disk used on your Windows 2000 or Windows Server 2003 host and, if it is a dynamic disk, change it to basic.

Caution: If you change a dynamic disk to a basic disk, you lose all data on the disk.

Use this procedure to convert a dynamic disk to a basic disk.

1. Open the disk management tool.

   Start > **Settings** > **Control Panel** > **Administrative Tools** > **Computer Management** > **Disk Management**
2. Delete all logical volumes on the disk. This destroys all data on the disk.
3. Right-click the **Disk <n>** button and select **Revert to Basic Disk**.
4. Create the partitions you want on the disk.

Configuring Dual- or Multiple-Boot SCSI Systems to Run with VMware GSX Server on a Linux Host

It may be possible to configure GSX Server so that you can use an operating system already installed and configured on a SCSI disk as a guest operating system inside a GSX Server virtual machine.

Using an existing physical SCSI disk — also called a SCSI raw disk — inside a virtual machine is supported only if the host has a BusLogic SCSI adapter. It may be possible to configure a host with a different SCSI adapter so the same operating system can be booted both natively and inside a virtual machine, but this approach is not supported by VMware. For details on some of the key issues involved, see Known Issues and Background Information on Using SCSI Physical Disks on page 217.

Before You Create the Virtual Machine Configuration

You must create a separate configuration for each guest operating system. Allow read and write access to the partitions used by that operating system only.

1. Before starting, you should read Setting Up Hardware Profiles in Virtual Machines on page 206 if you are running a Windows guest operating system. You should boot the guest operating system natively on the computer and create a hardware profile for the virtual machine before proceeding.
2. Check to see what SCSI ID is set for the drive you plan to use in the virtual machine.
3. Make certain that in addition to any SCSI drivers you have configured for the host, you have also installed the driver for a Mylex® (BusLogic) BT/KT-958 compatible host bus adapter. Drivers for BusLogic controllers are available from the LSI Logic Web site — *www.lsilogic.com*. Search the site for Mylex BusLogic BT/KT-958.

   The BusLogic driver needs to be installed in the profile for the guest operating system.

   Note: To use SCSI devices in a Windows XP virtual machine, you need a special SCSI driver available from the download section of the VMware Web site at *www.vmware.com/download*.
4. Check operating system partition mounts. Be sure the existing physical disk partitions that you plan to configure the virtual machine to use are not mounted by the Linux host.

   Caution: A physical disk partition should not be used (mounted) simultaneously by the host and the guest operating system. Because each operating system is

unaware of the other, data corruption may occur if both operating systems read or write to the same partition. It is critical that the virtual machine not be allowed to modify any partition mounted under the Linux host or in use by another virtual machine. To safeguard against this problem, be sure the partition you use for the virtual machine is not mounted under the Linux host.

5. Set the device group membership or device ownership. The master physical disk devices must be readable and writable by the user who runs GSX Server. On most distributions, the raw devices (such as `/dev/hda` and `/dev/hdb`) belong to group ID `disk`. If this is the case, you can add GSX Server users to the `disk` group. Another option is to change the owner of the device. Please think carefully about security issues when you explore different options here.

   It is typically a good idea to grant GSX Server users access to all `/dev/hd[abcd]` raw devices that contain operating systems or boot managers and then rely on GSX Server's physical disk configuration files to guard access. This provides boot managers access to configuration and other files they may need to boot the operating systems. For example, LILO needs to read `/boot` on a Linux partition to boot a non-Linux operating system that may be on another drive.

6. If you plan to run a second Linux installation from an existing partition as a guest operating system, and your physical machine's `/etc/lilo.conf` has a memory register statement such as `Append= "mem…"`, you may want to adjust the append memory parameter or create a new entry in LILO for running Linux in a virtual machine.

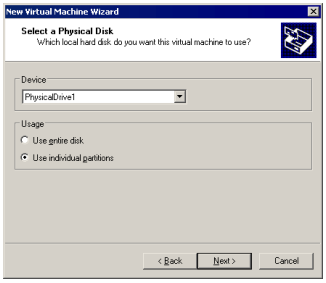
   Many newer Linux distributions recognize all physical memory in the physical machine, whereas many older Linux distributions see only the first 64MB of memory by default. Machines with more than 64MB of memory that run the older distributions may have the `Append= "mem=…"` parameter added under the `Image=…` section of `lilo.conf` to tell Linux to look for more memory than seen by default.

   If the amount of memory configured in `lilo.conf` exceeds the amount of memory assigned to the virtual machine, the guest operating system is likely to panic when the virtual machine tries to boot the second Linux installation.

   You can create another entry in `lilo.conf` for running Linux in a virtual machine by specifying a different amount of memory than what should normally be recognized when Linux boots directly on the physical machine.

Setting Up the Virtual Machine Configuration

1. Launch the VMware Virtual Machine Console.
2. Start the New Virtual Machine Wizard (**File** > **New** > **New Virtual Machine**) and select **Custom**.
3. When you reach the Select a Disk step, select **Use a physical disk**. The Select a Physical Disk screen appears.

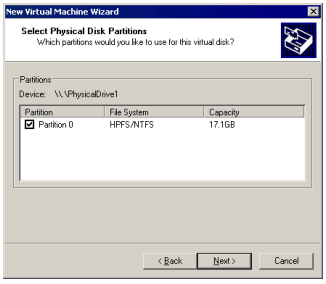


4. In the **Device** list, select the physical drive.

   Under **Usage**, select whether to use the entire disk or individual partitions.

   If you selected **Use entire disk**, click **Next** then go to step 5.

   If you selected **Use individual partitions**, the Select Physical Disk Partitions screen appears.



   Select the partitions you want the virtual machine to use, then click **Next**.

5. In the entry field, enter a name of your choice for the physical disk.

   Caution: If you browse to place the disk file in another directory, do not select an existing virtual disk file.

   To specify a device ID for the physical disk, click **Advanced**. In the **Virtual device node** list, select the SCSI ID that corresponds to the one used by your SCSI drive.

For example, if your SCSI drive has SCSI ID 2, select **SCSI 0:2**. If you do not know the SCSI ID set on your physical SCSI drive, try using **SCSI 0:0**.

On the advanced settings screen, you can also specify a disk mode. This is useful in certain special-purpose configurations in which you want to exclude disks from the snapshot. For more information on the snapshot feature, see Taking Snapshots on page 153.

Normal disks are included in the snapshot. In most cases, this is the setting you want.

Independent disks are not included in the snapshot. You have the following options for an independent disk:

- **Persistent** — changes are immediately and permanently written to the disk.
- **Nonpersistent** — changes to the disk are discarded when you power off or revert to the snapshot.

When you have set the filename and location you want to use and have made any selections you want to make on the advanced settings screen, click **Finish**.

6. Begin using your virtual machine.

Known Issues and Background Information on Using SCSI Physical Disks

Some known issues with SCSI physical disks involve disk geometry, appropriate drivers and operating system configuration.

Geometry

In some cases, it is not possible to boot a raw SCSI drive inside a virtual machine because the SCSI adapter in the physical computer and the BusLogic adapter in the virtual machine describe the drive in different ways. The virtual machine might hang during the boot, GSX Server might crash or GSX Server might fail with an ASSERT or other error message.

This problem is most likely to affect smaller drives — less than 2GB.

In order to share the same BIOS interface used by IDE disks (which is required in order to boot), all SCSI disks need to have a geometry, which is a fabricated value for the number of cylinders, sectors and heads on the disk.

In fact, a SCSI disk appears to a computer as a single flat entity from sector 1 up to the highest sector on the disk. As a result, every SCSI vendor has its own approach to taking the capacity of a SCSI disk and generating a geometry to use for booting.

The conversion from a given geometry to an absolute sector number depends on the geometry. If you have a disk with a boot sector written by a program running on the

host and you try to boot that disk inside a virtual machine, the boot program can fail if the host geometry does not match the geometry used by the BusLogic virtual SCSI adapter. The symptoms are that you see the first part of the boot loader — possibly an `LI` from LILO, for example — but then the boot either stops or crashes.

BusLogic uses the following rules for generating disk geometries:

Disk size	Heads	Sectors
<= 1GB	64	32
> 1GB and <= 2GB	128	32
> 2GB	255	63

In each case the number of cylinders is calculated by taking the total capacity of the disk and dividing by (heads*sectors). Fortunately, for sufficiently big disks, practically all vendors use 255 heads and 63 sectors.

Drivers

In contrast to IDE adapters, SCSI adapters are not interchangeable and cannot all use the same drivers. That is, if you have an Adaptec SCSI host adapter in your machine and you remove it and replace it with a BusLogic SCSI host adapter, your operating system is likely not to boot unless you install a BusLogic driver.

Dual booting from a disk that is also used as a virtual disk is no different. To your operating system, it appears that the SCSI card in the machine suddenly changed from whatever you own to a BusLogic card, and your operating system needs to have a valid BusLogic driver installed. If that driver is not installed, you get a panic, a blue screen or some similar fatal error as soon as the boot process tries to switch from the BIOS bootstrap to the disk driver installed in the operating system.

Operating System Configuration

Many operating systems have configuration information that is different for SCSI and IDE drives. For example, Linux uses `/dev/hd[x]` as the device name for IDE disks and `/dev/sd[x]` for SCSI disks. References to these names appear in `/etc/fstab` and other configuration files.

This is one reason that booting a raw IDE disk as a SCSI disk or vice versa does not work well (if at all).

However, even when you are dealing only with SCSI devices, it is possible for an operating system to encode information in a way that causes problems when you are dual booting. For example, Solaris names its SCSI disks `/dev/c[x]t[y]d[z]s0`, where the `y` represents the SCSI ID. So if you had a physical disk configured as SCSI ID

3 on the host and as SCSI ID 0 in your virtual machine's configuration file, it would move if you were running Solaris, and most likely Solaris would not boot.

The precise dependencies in various operating systems can be complex. That is why it is safest to configure SCSI physical disks in a virtual machine using the same SCSI ID as they use on the host.

Installing an Operating System onto a Physical Partition from a Virtual Machine

In some situations, you may want to install a guest operating system directly on a physical disk or partition — also known as a raw disk — even if you do not need to boot that disk on the host, outside of the virtual machine.

It is possible to use either an unused partition or a completely unused disk on the host as a disk in the virtual machine. However, it is important to be aware that an operating system installed in this setting probably cannot boot outside of the virtual machine, even though the data is available to the host.

Caution: You cannot use a physical disk that is stored on a SAN. You must use a disk or a partition on the GSX Server host.

If you have a dual-boot system and want to configure a virtual machine to boot from an existing partition, see Configuring a Dual-Boot Computer for Use with a Virtual Machine on page 198. The instructions in this section do not apply to a disk with a previously installed operating system.

Caution: Physical disks are an advanced feature and should be configured only by advanced users.

GSX Server uses description files to control access to each physical disk on the system. These description files contain access privilege information that controls a virtual machine's access to certain partitions on the disks. This mechanism prevents users from accidentally running the host operating system again as a guest or running a guest operating system that the virtual machine is not configured to use. The description file also prevents accidental writes to physical disk partitions from badly behaved operating systems or applications.

Use the New Virtual Machine Wizard to configure a virtual machine to use existing physical disk partitions. The wizard guides you though creating a new virtual machine including configuring the physical disk description files. Rerun the wizard to create a separate configuration for each guest operating system installed on a raw partition.

Note: While installing the guest operating system on a physical disk, if your virtual machine does not boot from the CD-ROM, try changing the boot order in the virtual machine's BIOS. Restart the virtual machine, then press F2 while the virtual machine is booting to enter the BIOS. Change the boot order there.

Read the section appropriate to your GSX Server host operating system.

- Configuring a Windows Host on page 221
- Configuring a Linux Host on page 223

Configuring a Windows Host

The following sections describe configuring physical disks on a Windows host.

Windows 2000, Windows XP and Windows Server 2003 Dynamic Disks

If your host is running Windows 2000, Windows XP or Windows Server 2003 and is using dynamic disks, see Do Not Use Windows 2000, Windows XP and Windows Server 2003 Dynamic Disks as Physical Disks on page 213.

Configuring the Virtual Machine to Use a Physical Disk

Use the following steps to run a guest operating system from a physical disk.

Note: If you use a Windows host's IDE disk in a physical disk configuration, it cannot be configured as the slave on the secondary IDE channel if the master on that channel is a CD-ROM drive.

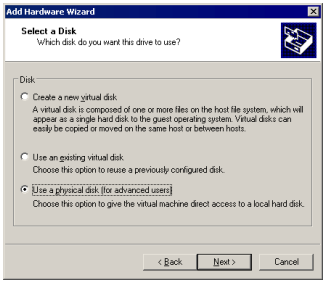
1. Identify the raw partition on which you plan to install the guest operating system.

   Check the guest operating system documentation regarding the type of partition on which the operating system can be installed. For example, operating systems like DOS, Windows 95 and Windows 98 must be installed on the first primary partition while others, like Linux, can be installed on a primary or extended partition on any part of the drive.

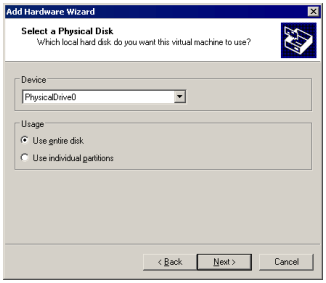
   Identify an appropriate raw partition or disk for the guest operating system to use. Be sure that the raw partition is not mounted by the Windows host and not in use by others. Also, be sure the raw partition or disk does not have data you will need in the future; if it does, back up that data now.

2. Start the New Virtual Machine Wizard (**File** > **New** > **New Virtual Machine**) and select **Custom**.

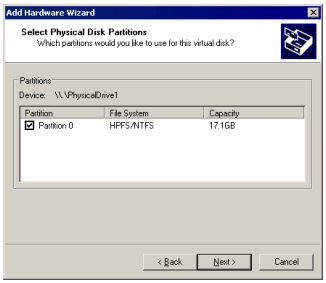
3. When you reach the Select a Disk step, select **Use a physical disk**.


Add Hardware Wizard
Select a Disk
Which disk do you want this drive to use?
Disk
Create a new virtual disk
A virtual disk is composed of one or more files on the host file system, which will appear as a single hard disk to the guest operating system. Virtual disks can easily be copied or moved on the same host or between hosts.
Use an existing virtual disk
Choose this option to reuse a previously configured disk.
Use a physical disk (for advanced users)
Choose this option to give the virtual machine direct access to a local hard disk.
< Back
Next >
Cancel

4. Choose the physical hard disk to use from the drop-down list. Select whether you want to use the entire disk or use only individual partitions on the disk. Click **Next**.



5. If you selected **Use individual partitions** in the previous step, select which partitions you want to use in the virtual machine. If you selected **Use entire disk**, this step does not appear.



Click **Next**.

6. The partition on which you are installing the guest operating system should be unmapped in the host.

Caution: Corruption is possible if you allow the virtual machine to modify a partition that is simultaneously mounted under Windows. Since the virtual machine and guest operating system access a physical disk partition while the host continues to run Windows, it is critical that you not allow the virtual machine to modify any partition mounted by the host or in use by another virtual machine. To safeguard against this problem, be sure the physical disk partition you use for the virtual machine is not in use by the host.

Use Disk Management (**Start** > **Settings** > **Control Panel** > **Administrative Tools** > **Computer Management** > **Storage** > **Disk Management**). Select the partition you want to unmap, then choose **Action** > **All Tasks** > **Change Drive Letter and Path**. Click the **Remove** button.

7. Use the virtual machine settings editor (**VM** > **Settings**) if you want to change any configuration options from the wizard defaults — for example, to change the amount of memory allocated to the virtual machine.
8. At this point you are ready to begin installing the guest operating system onto the physical disk you configured for the virtual machine. For more details, read the installation notes for various guest operating systems in the *VMware Guest Operating System Installation Guide*, available from the VMware Web site or from the Help menu.

Configuring a Linux Host

1. Identify the raw partition on which the guest operating system will be installed.

   Check the guest operating system documentation regarding the type of partition on which the operating system can be installed. For example, operating systems like DOS, Windows 95 and Windows 98 must be installed on the first primary partition while others, like Linux, can be installed on a primary or extended partition on any part of the drive.

   Identify an appropriate raw partition or disk for the guest operating system to use. Check that the raw partition is not mounted by the Linux host and not in use by others. Also, be sure the raw partition or disk does not have data you will need in the future; if it does, back up that data now.
2. Check the operating system partition mounts. Be sure the existing disk partitions that you plan to use in the virtual machine are not mounted by Linux.
3. Set the device group membership or device ownership.

   The master physical disk device or devices need to be readable and writable by the user who runs GSX Server. On most distributions, the raw devices, such as `/dev/hda` (IDE physical disk) and `/dev/sdb` (SCSI physical disk) belong to

group ID `disk`. If this is the case, you can add GSX Server users to the `disk` group. Another option is to change the owner of the device. Please think carefully about security issues when you explore different options here.

It is a good idea to grant GSX Server users access to all `/dev/hd[abcd]` raw devices that contain operating systems or boot managers, then rely on GSX Server's physical disk configuration files to guard access. This provides boot managers access to configuration and other files they may need to boot the operating systems. For example, LILO needs to read `/boot` on a Linux partition to boot a non-Linux operating system that may be on another drive.

4. Start the New Virtual Machine Wizard (**File** > **New** > **New Virtual Machine**) and select **Custom**.
5. When you reach the Select a Disk step, select **Use a physical disk**.
6. If the physical disk you plan to use has multiple partitions on it already, be aware that certain operating systems (DOS, Windows 95, Windows 98) must be installed on the first primary partition.

   Caution: Corruption is possible if you allow the virtual machine to modify a partition that is simultaneously mounted under the Linux host operating system. Since the virtual machine and guest operating system access an existing partition while the host continues to run Linux, it is critical that the virtual machine not be allowed to modify any partition mounted by the host or in use by another virtual machine.

   To safeguard against this problem, be sure the partition you use for the virtual machine is not mounted under the Linux host.
7. At this point you are ready to begin installing the guest operating system on the physical disk you configured for the virtual machine. For more details, read the installation notes for various guest operating systems in the *VMware Guest Operating System Installation Guide*, available from the VMware Web site or from the Help menu.

Disk Performance in Windows NT Guests on Multiprocessor Hosts

Some users have seen slower than expected disk input and output performance when running Windows NT guest operating systems. They see the problem in GSX Server virtual machines using IDE virtual disks on multiprocessor host computers. The I/O issue is especially noticeable when the virtual machine is booting.

Note: Performance in Windows NT guest operating systems may also be affected by disk fragmentation on the host computer. For details, see Configuring and Maintaining the Host Computer in the *VMware GSX Server Administration Guide*.

Improving Performance

You may increase performance by enabling DMA (direct memory access) on the virtual hard disk's IDE channel in the virtual machine.

If you have a virtual disk and a DVD/CD-ROM attached as master and slave to the primary IDE controller (channel 0) and you want to enable DMA, power off the virtual machine and use the virtual machine settings editor (**VM** > **Settings**) to move the DVD/CD-ROM drive to the secondary IDE controller (channel 1) at IDE 1:0.

You can enable the DMA feature after you finish installing Windows NT. You must install Service Pack 3 or higher in the virtual machine to enable this option.

Once the virtual machine is running Windows NT, insert an SP3 or SP4 CD in the drive and run `DMACHECK.EXE` from the `\SUPPORT\UTILS\I386` folder on the CD. Or download `DMACHECK.EXE` from the Microsoft Web site (*support.microsoft.com/support/kb/articles/Q191/7/74.ASP*).

Click the **Enabled** option for the IDE controller and channel configured for the virtual disk. Typically, this is channel 0 only, unless you have the virtual machine configured with multiple virtual disks and no virtual DVD/CD-ROM drive.

As noted above, you should not enable DMA on an IDE channel with a virtual DVD/CD-ROM drive attached.

CHAPTER 7

Networking

VMware GSX Server provides virtual networking components that let you create a wide range of configurations.

If you create a virtual machine with the New Virtual Machine Wizard, the wizard lets you choose any of the common configurations — bridged networking, network address translation (NAT) and host-only networking. The wizard then connects the virtual machine to the appropriate virtual network.

If you create a virtual machine with the VMware Management Interface, the wizard sets up bridged networking for the virtual machine. You can change the type of networking to network address translation (NAT) or host-only networking after you create the virtual machine.

You can set up more specialized configurations by choosing the appropriate settings in the virtual machine settings editor, in the Virtual Network Editor (on Windows hosts) and on your host computer.

On a Windows host, the software needed for all networking configurations is installed when you install GSX Server. On a Linux host, all components are available if you choose to have both bridged and host-only networking available to your virtual machines at the time you install GSX Server.

The first topics in this section give you a quick look at the virtual networking components that GSX Server provides and show how you can use them with your virtual machine. The rest of the section provides more detail on some networking capabilities and specialized configurations.

- Components of the Virtual Network on page 229
- Common Networking Configurations on page 231
- Custom Networking Configurations on page 235
- Changing the Networking Configuration on page 238
- Advanced Networking Topics on page 250
- Understanding NAT on page 269
- Using Samba for File Sharing on a Linux Host on page 280

Components of the Virtual Network

The following items are components of a virtual network:

Virtual switch — Like a physical switch, a virtual switch lets you connect other networking components together. Virtual switches are created as needed by the GSX Server software, up to a total of ten switches on a Windows host or 100 switches on a Linux host. You can connect one or more virtual machines to a switch.

A few of the switches and the networks associated with them are, by default, used for special named configurations. The bridged network normally uses VMnet0. The host-only network uses VMnet1 by default. And the NAT network uses VMnet8 by default. The others available networks are simply named VMnet2, VMnet3, VMnet4, and so on.

You connect a virtual machine to a switch by selecting the virtual network adapter you want to connect in the virtual machine settings editor, then configuring it to use the desired virtual network.

Bridge — The bridge lets you connect your virtual machine to the LAN used by your host computer. It connects the virtual network adapter in your virtual machine to the physical Ethernet adapter in your host computer.

The bridge is installed during GSX Server installation (on a Linux host, you must choose to make bridged networking available to your virtual machines). It is set up automatically when you create a new virtual machine using bridged networking.

Additional virtual bridges can be set up for use in custom configurations that require connections to more than one physical Ethernet adapter on the host computer.

Host virtual adapter — The host virtual adapter is a virtual Ethernet adapter that appears to your host operating system as a VMware virtual Ethernet adapter on a Windows host and as a host-only interface on a Linux host. It allows you to communicate between your host computer and the virtual machines on that host computer. The host virtual adapter is used in host-only and NAT configurations.

The host virtual adapter is not connected to any external network unless you set up special software on the host computer — a proxy server, for example — to connect the host-only adapter to the physical network adapter.

The software that creates the host virtual adapter is installed when you install GSX Server (on a Linux host, you must choose to make host-only networking available to your virtual machines). A host virtual adapter is then created automatically when you boot the host computer.

You can set up additional host virtual adapters as needed.

NAT device — The NAT (network address translation) device allows you to connect your virtual machines to an external network when you have only one IP network address on the physical network, and that address is used by the host computer. You can, for example, use NAT to connect your virtual machines to the Internet through a dial-up connection on the host computer or through the host computer's Ethernet adapter or wireless Ethernet adapter. NAT is also useful when you need to connect to a non-Ethernet network, such as Token Ring or ATM.

The NAT device is set up automatically when you install GSX Server. (On a Linux host, you must choose to make NAT available to your virtual machines.)

DHCP server — The DHCP (dynamic host configuration protocol) server provides IP network addresses to virtual machines in configurations that are not bridged to an external network — for example, host-only and NAT configurations.

Network adapter — One virtual network adapter is set up for your virtual machine when you create it with the New Virtual Machine Wizard using any type of networking (a virtual network adapter is always added to a virtual machine created with the VMware Management Interface). It appears to the guest operating system as an AMD PCNET PCI adapter.

You can create and configure up to four virtual network adapters in each virtual machine using the virtual machine settings editor.

The adapter can use one of two drivers. You can choose between the `vlance` driver, which installs automatically, and the `vmxnet` driver, which provides better network performance. The difference in network performance is most noticeable if the virtual machine is connected to a Gigabit Ethernet card on the host.

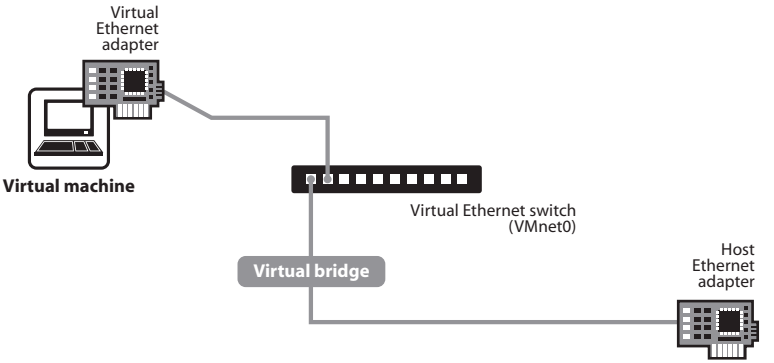
Common Networking Configurations

The following sections illustrate the networking configurations that are set up for you automatically when you choose the standard networking options in the New Virtual Machine Wizard or virtual machine settings editor.

- Bridged Networking on page 231
- Network Address Translation (NAT) on page 232
- Host-Only Networking on page 234

Only one virtual machine is shown in each example, but multiple virtual machines can be connected to the same virtual Ethernet switch. On a Windows host, you can connect an unlimited number of virtual network devices to a virtual switch. On a Linux host, you can connect up to 32 devices.

Bridged Networking



Bridged networking connects a virtual machine to a network using the host computer's Ethernet adapter.

Bridged networking is set up automatically if you select **Use bridged networking** in the New Virtual Machine Wizard or if you select the **Typical** setup path. This selection is available on a Linux host only if you enable the bridged networking option when you install GSX Server.

If your host computer is on an Ethernet network, bridged networking is often the easiest way to give your virtual machine access to that network. On a Windows host, you can use bridged networking to connect to either a wired or a wireless network. On a Linux host, you can use bridged networking to connect to a wired network.

If you use bridged networking, your virtual machine needs to have its own identity on the network. For example, on a TCP/IP network, the virtual machine needs its own IP

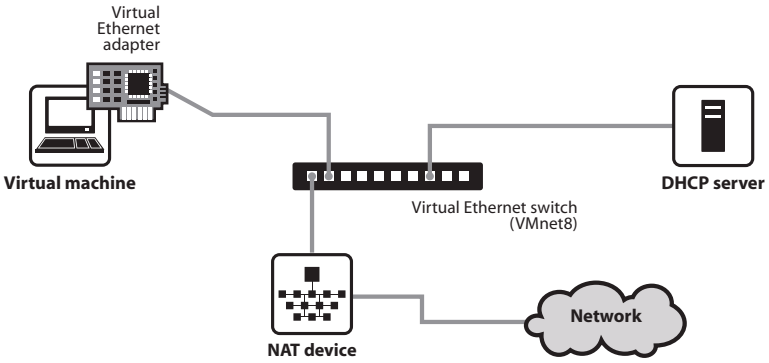
address. Your network administrator can tell you whether IP addresses are available for your virtual machine and what networking settings you should use in the guest operating system. Generally, your guest operating system may acquire an IP address and other network details automatically from a DHCP server, or you may need to set the IP address and other details manually in the guest operating system.

If you use bridged networking, the virtual machine is a full participant in the network. It has access to other machines on the network and can be contacted by other machines on the network as if it were a physical computer on the network.

Be aware that if the host computer is set up to boot multiple operating systems and you run one or more of them in virtual machines, you need to configure each operating system with a unique network address. People who boot multiple operating systems often assign all systems the same address, since they assume only one operating system will run at a time. If you use one or more of the operating systems in a virtual machine, this assumption is no longer true.

If you make some other selection in the New Virtual Machine Wizard and later decide you want to use bridged networking, you can make that change in the virtual machine settings editor (**VM** > **Settings**). For details, see Changing the Networking Configuration on page 238.

Network Address Translation (NAT)



NAT gives a virtual machine access to network resources using the host computer's IP address.

A network address translation connection is set up automatically if you follow the **Custom** path in the New Virtual Machine Wizard and select **Use network address translation**.

If you want to connect to the Internet or other TCP/IP network using the host computer's dial-up networking or broadband connection and you are not able to give your virtual machine an IP address on the external network, NAT is often the easiest way to give your virtual machine access to that network.

NAT also allows you to connect to a TCP/IP network using a Token Ring adapter on the host computer.

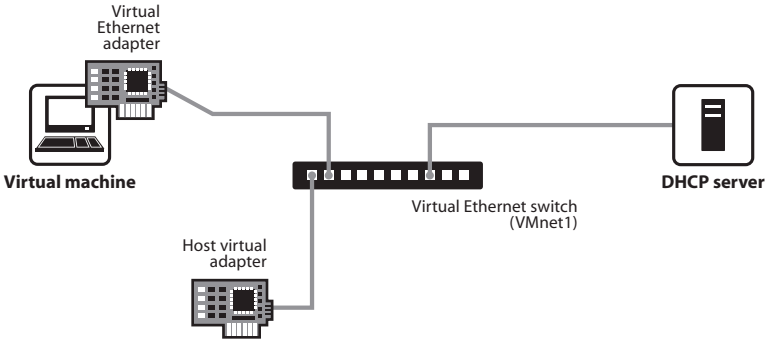
If you use NAT, your virtual machine does not have its own IP address on the external network. Instead, a separate private network is set up on the host computer. Your virtual machine gets an address on that network from the VMware virtual DHCP server. The VMware NAT device passes network data between one or more virtual machines and the external network. It identifies incoming data packets intended for each virtual machine and sends them to the correct destination.

If you select NAT, the virtual machine can use many standard TCP/IP protocols to connect to other machines on the external network. For example, you can use HTTP to browse Web sites, FTP to transfer files and Telnet to log on to other computers. In the default configuration, computers on the external network cannot initiate connections to the virtual machine. That means, for example, that the default configuration does not let you use the virtual machine as a Web server to send Web pages to computers on the external network.

If you make some other selection in the New Virtual Machine Wizard and later decide you want to use NAT, you can make that change in the virtual machine settings editor (**VM** > **Settings**). For details, see Changing the Networking Configuration on page 238.

For a more thorough discussion of NAT, see Understanding NAT on page 269.

Host-Only Networking



Host-only networking creates a network that is completely contained within the host computer.

A host-only network is set up automatically if you select **Use Host-Only Networking** in the New Virtual Machine Wizard. On Linux hosts, this selection is available only if you enabled the host-only networking option when you installed GSX Server.

Host-only networking provides a network connection between the virtual machine and the host computer, using a virtual Ethernet adapter that is visible to the host operating system. This approach can be useful if you need to set up an isolated virtual network.

If you use host-only networking, your virtual machine and the host virtual adapter are connected to a private TCP/IP network. Addresses on this network are provided by the VMware DHCP server.

If you make some other selection in the New Virtual Machine Wizard and later decide you want to use host-only networking, you can make that change in the virtual machine settings editor (**VM** > **Settings**). For details, see Changing the Networking Configuration on page 238.

Routing and Connection Sharing

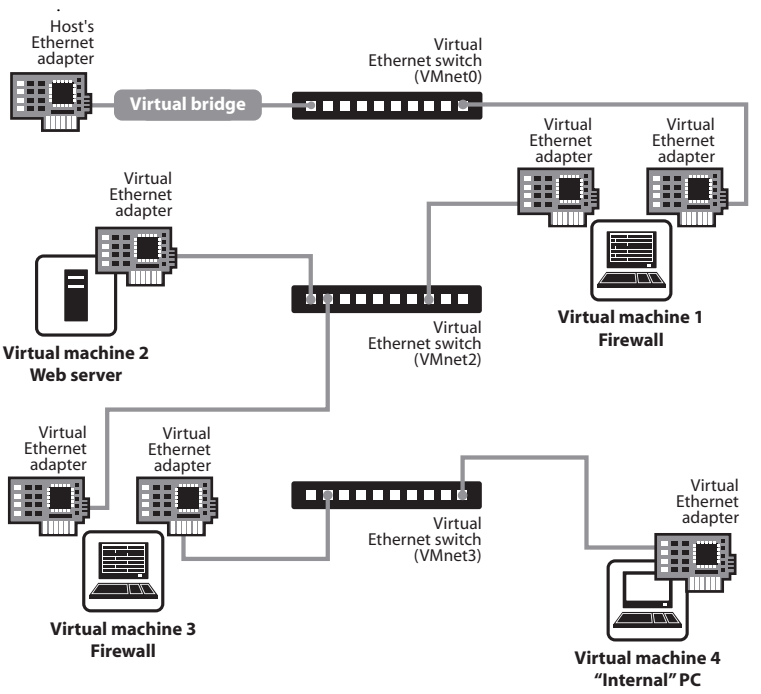
- If you install the proper routing or proxy software on your host computer, you can establish a connection between the host virtual Ethernet adapter and a physical network adapter on the host computer. This allows you, for example, to connect the virtual machine to a Token Ring or other non-Ethernet network.
- On a Windows 2000 or Windows Server 2003 host computer, you can use host-only networking in combination with the Internet connection sharing feature in Windows to allow a virtual machine to use the host's dial-up networking adapter or other connection to the Internet. See your Windows documentation for details on configuring Internet connection sharing.

Custom Networking Configurations

The virtual networking components provided by GSX Server make it possible for you to create sophisticated virtual networks. The virtual networks can be connected to one or more external networks, or they may run entirely on the host computer.

Setting up networking components for your custom virtual network is a straightforward process. Before attempting to set up complex virtual networks, you should have a good understanding of how to configure network devices in your host and guest operating systems.

The sample configuration described in this section illustrates many of the ways you can combine devices on a virtual network. Other custom configurations are described in Advanced Networking Topics on page 250 and Understanding NAT on page 269.



In this custom configuration, a Web server connects through a firewall to an external network. An administrator's computer can connect to the Web server through a second firewall.

To set up this configuration, you must create four virtual machines and use the virtual machine settings editor to adjust the settings for their virtual Ethernet adapters. You also need to install the appropriate guest operating systems and application software in each virtual machine and make the appropriate networking settings in each virtual machine.

1. Set up four virtual machines using the New Virtual Machine Wizard.

   Create the first virtual machine with bridged networking so it can connect to an external network using the host computer's Ethernet adapter.

   Create the other three virtual machines without networking. You will set up their virtual Ethernet adapters in later steps.

2. Launch a VMware Virtual Machine Console and open virtual machine 1. Do not power on the virtual machine.

   Use the virtual machine settings editor (**VM** > **Settings**) to add a second virtual network adapter, as described in Changing the Networking Configuration on page 238. Connect the second adapter to **Custom (VMnet2)**.

   Click **OK** to save the configuration and close the virtual machine settings editor.

3. If a console is not running, launch one. Open virtual machine 2. Do not power on the virtual machine.

   Use the virtual machine settings editor (**VM** > **Settings**) to add a virtual network adapter. Connect the adapter to **Custom (VMnet2)**.

   Click **OK** to save the configuration and close the virtual machine settings editor.

4. If a console is not running, launch one. Open virtual machine 3. Do not power on the virtual machine.

   Use the virtual machine settings editor to add a virtual network adapter. Connect the adapter to **Custom (VMnet2)**.

   Use the virtual machine settings editor to add a second virtual network adapter. Connect the adapter to **Custom (VMnet3)**.

   Click **OK** to save the configuration and close the virtual machine settings editor.

5. If a console is not running, launch one. Open virtual machine 4. Do not power on the virtual machine.

   Use the virtual machine settings editor to add a virtual network adapter. Connect the adapter to **Custom (VMnet3)**.

   Click **OK** to save the configuration and close the virtual machine settings editor.

6. Determine the network addresses used for VMnet2 and VMnet3.

Note: On a Windows host, you may skip the steps for configuring network addresses manually and, instead, use GSX Server's DHCP server. Choose **Host** > **Virtual Network Settings** > **DHCP** and add VMnet2 and VMnet3 to the list of virtual networks served by the virtual DHCP server. Then skip to step 9.

On a Windows host, open a command prompt on the host computer and run `ipconfig /all`. Note the network addresses used by each virtual adapter.

On a Linux host, run `ifconfig` at the console or in a terminal window on the host computer. Note the network addresses used by each virtual switch.

7. Launch a console, open each virtual machine in turn and install the appropriate guest operating system.
8. Configure the networking in each guest operating system.

   For the bridged Ethernet adapter in virtual machine 1, use the networking settings needed for a connection to the external network. If the virtual machine gets its IP address from a DHCP server on the external network, the default settings should work.

   For the second Ethernet adapter in virtual machine 1, manually assign an IP address in the range you are using with VMnet2.

   In virtual machine 2, assign an IP address in the range you are using with VMnet2.

   In virtual machine 3, network adapters are connected to VMnet2 and VMnet3. Assign each adapter an IP address in the range you are using with the virtual network to which it is connected.

   In virtual machine 4, assign an IP address in the range you are using with VMnet3.
9. Install the necessary application software in each virtual machine.

Changing the Networking Configuration

Using the virtual machine settings editor (**VM** > **Settings**), you can change the configuration of your virtual networks by

- Adding and Modifying Virtual Network Adapters on page 238
- Configuring Bridged Networking Options on a Windows Host on page 242
- Enabling, Disabling, Adding and Removing Host Virtual Adapters on page 247

Adding and Modifying Virtual Network Adapters

You can add new or configure existing virtual network adapters from the VMware Virtual Machine Console and from the VMware Management Interface. If VirtualCenter manages the virtual machines on your GSX Server host, use the management interface to add a new or configure an existing virtual network adapter.

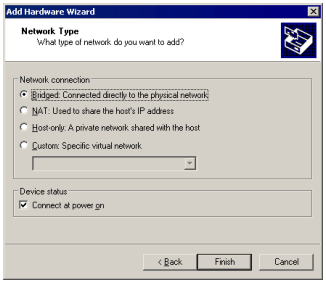
The settings you can configure include the virtual network device to which the virtual machine is bound and the network driver it uses.

When choosing the network driver for a virtual network adapter, you can choose between the `vlance` driver, which installs automatically, and the `vmxnet` driver, which provides better network performance. The difference in network performance is most noticeable if the virtual machine is connected to a Gigabit Ethernet card.

Adding and Configuring a Virtual Network Adapter from the Console

To add a new virtual network adapter to a virtual machine, make sure the virtual machine is powered off, then complete the following steps.

1. Open the virtual machine settings editor. Choose **VM** > **Settings**.
2. Click **Add**.
3. The Add Hardware Wizard starts. Select **Ethernet Adapter**. Click **Next**. The Network Type screen appears.


Add Hardware Wizard
Network Type
What type of network do you want to add?
Network connection
Bridged: Connected directly to the physical network
NAT: Used to share the host's IP address
Host-only: A private network shared with the host
Custom: Specific virtual network
Device status
Connect at power on
< Back
Finish
Cancel

4. Specify the type of networking this virtual NIC uses. Select **Bridged**, **NAT**, **Host-only**, **Custom** or **Named**.

   If you select **Custom**, choose the VMnet virtual network you want to use for the network from the drop-down list.

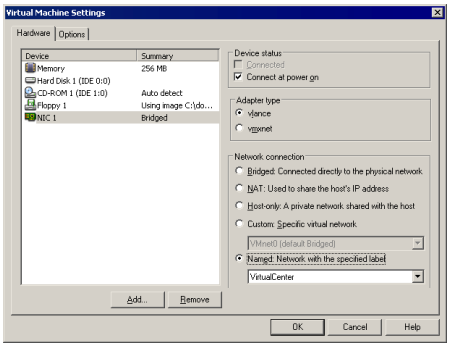
   Note: Although VMnet0, VMnet1 and VMnet8 are available in this list, they are normally used for bridged, host-only and NAT configurations, respectively. Special steps are required to make them available for use in custom configurations. You should choose one of the other switches.

   Note: If the virtual machine is managed by VirtualCenter, select **Named** and choose a network from the list.

5. Click **Finish**. The new adapter is added.
6. Click **OK** to save your configuration and close the virtual machine settings editor.

To change the configuration of an existing virtual network adapter, follow these steps.

1. Open the virtual machine settings editor. Choose **VM** > **Settings**.
2. Select the adapter you want to modify.



3. Specify the type of networking this virtual NIC uses. Select **Bridged**, **NAT**, **Host-only**, **Custom** or **Named**.

   If you select **Custom**, choose the VMnet virtual network you want to use for the network from the drop-down list.

   Note: If the virtual machine is managed by VirtualCenter, select **Named** and choose a network from the list.

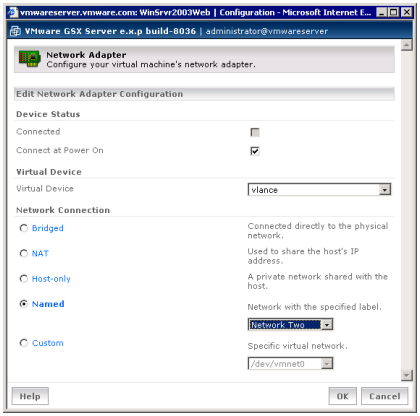
4. Click **OK** to save your changes and close the virtual machine settings editor.
5. Be sure the guest operating system is configured to use an appropriate IP address on the new network. If the guest is using DHCP, release and renew the

lease. If the IP address is set statically, be sure the guest has an address on the correct virtual network.

Adding and Configuring a Virtual Network Adapter from the Management Interface

To add a new virtual network adapter to a virtual machine, make sure the virtual machine is powered off, then complete the following steps.

1. On the Hardware page, click **Add Device**. The Add Device Wizard starts.
2. Click **Network Adapter**. The Network Adapter page appears.



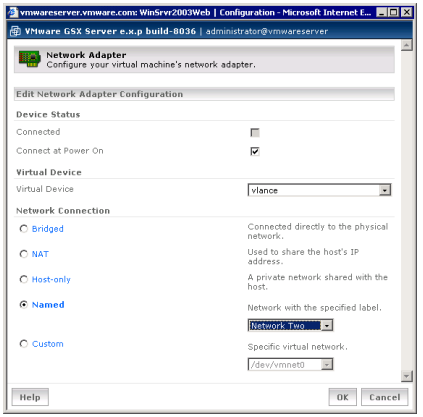
3. To connect the virtual NIC when the virtual machine is powered on, check **Connect at Power On**.
4. In the **Virtual Device** list, select the network driver you want the virtual machine to use. Choose either the `vlance` or `vmxnet` driver.
5. Specify the type of networking this virtual NIC uses. Select **Bridged**, **NAT**, **Host-only**, **Custom** or **Named**.

   If you selected a custom network, select the specific virtual network from the list.

   Note: If the virtual machine is managed by VirtualCenter, select **Named** and choose a network from the list.
6. Click **OK** to add the network adapter.

To change the configuration of an existing virtual network adapter, complete the following steps.

1. In the Hardware page, under **Network Adapter**, click **Edit**. The Network Adapter page appears.



2. To connect the virtual network adapter when the virtual machine is powered on, check **Connect at Power On**.
3. In the **Virtual Device** list, select the network driver you want the virtual machine to use. Choose either the `vlance` or `vmxnet` driver.
4. Specify the type of networking this virtual NIC uses. Select **Bridged**, **NAT**, **Host-only**, **Custom** or **Named**.

   If you selected a custom network, select the specific virtual network from the list.

   Note: If the virtual machine is managed by VirtualCenter, select **Named** and choose a network from the list.
5. Click **OK** to save your changes and close the window.

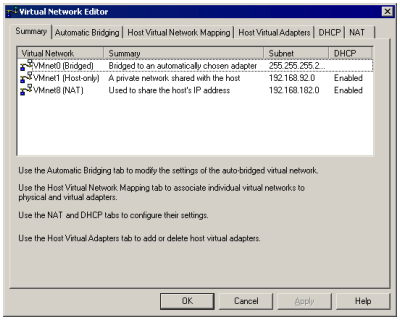
Configuring Bridged Networking Options on a Windows Host

You can view and change the settings for bridged networking on your host. These changes affect all virtual machines using bridged networking on the host.

You can decide which network adapters on your host to use for bridged networking. You can map specific network adapters to specific virtual networks (VMnets).

1. Launch a VMware Virtual Machine Console.
2. Choose **Host** > **Virtual Network Settings**.

   The Virtual Network Editor appears, with the Summary tab active.



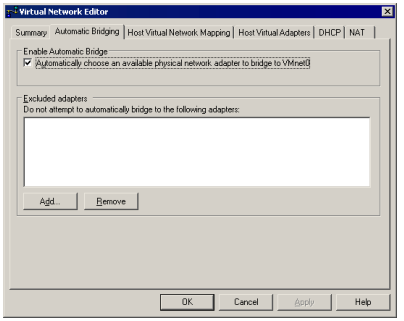
3. By default, the VMnet0 virtual network is set up in bridged mode and bridges to one of the active Ethernet adapters on the host computer.

   The choice of which adapter it uses is arbitrary. It is a good idea to let GSX Server choose an available physical network adapter for bridging, as it provides fault tolerance. If a network adapter becomes unavailable (for example, if it is unplugged or removed from the host), the network bridge automatically switches to another network adapter on the host.

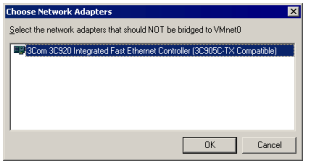
   You can restrict the range of choices using options on the Automatic Bridging tab.

   (Also shown are VMnet1, the default virtual network for host-only networking, and VMnet8, the default virtual network for NAT, if they are enabled in GSX Server.)

4. To exclude one or more physical Ethernet adapters from the list to which VMnet0 may be bridged, click the **Automatic Bridging** tab.

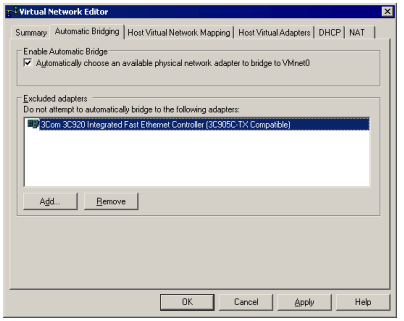


To exclude an Ethernet adapter, click **Add** to add it to the list of excluded devices.


Choose Network Adapters
Select the network adapters that should NOT be bridged to VMnet0
3Com 3C920 Integrated Fast Ethernet Controller (3C905C-TX Compatible)
OK
Cancel

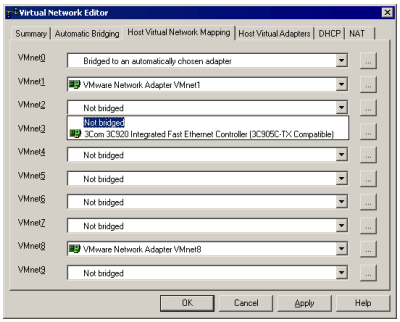
In the Choose Network Adapters dialog box, select the listing for the adapter you want to exclude, then click **OK**.

To remove an adapter from the list of excluded adapters, select its name in the list, then click **Remove**.



If you are using teamed network adapters on your host, you can exclude the physical network adapters from bridged networking. For information about teamed network adapters, see Configuring Bridged Networking When Using Teamed Network Interface Cards on Your Host on page 259.

5. To designate a physical Ethernet adapter to be used for bridged networking on virtual switches named VMnet2–VMnet7, click the **Host Virtual Network Mapping** tab.

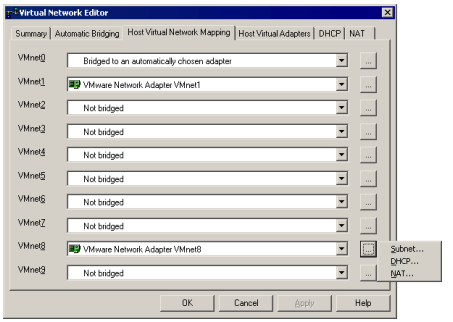


Choose an adapter from the drop-down list beside the name of the virtual switch you want to use.

If you are using teamed network adapters on your host, you can choose the teamed NIC for VMnet0.

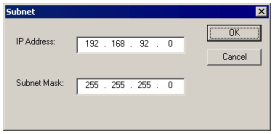
Caution: Be careful when you change the bridged adapter mappings. If you re-assign a physical Ethernet adapter to a different virtual network, any virtual machine using the original network loses its network connectivity via that network. You must then change the setting for each affected virtual machine's network adapter individually. This can be especially troublesome if your host has only one physical Ethernet adapter and you reassign it to a VMnet other than VMnet0; even though the VMnet still appears to be bridged to an automatically chosen adapter, the only adapter it can use has been assigned to another VMnet.

6. To make changes to the subnet or the DHCP settings for a virtual network, click the button on the right that corresponds to the virtual network you want to configure, then choose **Subnet** or **DHCP**.



Changing the Subnet

In the Subnet dialog box, you can change the subnet's IP address and the subnet mask.



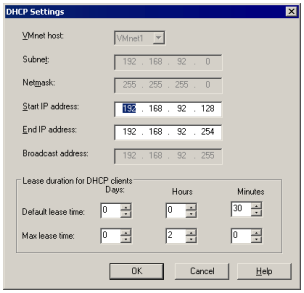
The address should specify a valid network address that is suitable for use with the subnet mask.

The default subnet mask is 255.255.255.0 (a class-C network). Typically, this means you should modify only the third number in the IP address — for example, x in 192.168.x.0 or 172.16.x.0. In general, you should not change the subnet mask. Certain virtual network services may not work as well with a customized subnet mask.

When you modify the network address or subnet mask, GSX Server automatically updates the IP address settings for other components — such as DHCP, NAT and host virtual adapter — on that virtual network to reflect the new settings. The specific settings that are automatically updated include DHCP lease range, DHCP server address, NAT gateway address and host virtual adapter IP address. However, if you have changed any of these settings from its default value — even if you have later changed the setting back to the default — GSX Server does not update that setting automatically. It presumes that custom settings are not to be modified.

Changing DHCP Settings

In the DHCP settings dialog box, you can change the range of IP addresses provided by the DHCP server on a particular virtual network. You can also set the duration of leases provided to clients on the virtual network.



7. When you have made all the changes you want to make on all tabs of the VMware Network Editor, click **OK**.

Enabling, Disabling, Adding and Removing Host Virtual Adapters

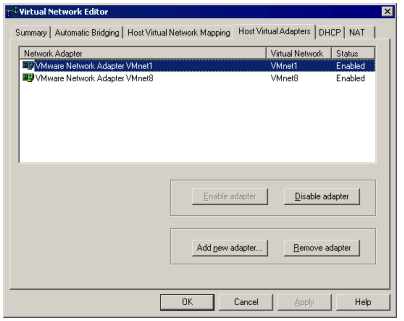
When you install GSX Server, two network adapters are added to the configuration of your host operating system — one that allows the host to connect to the host-only network and one that allows the host to connect to the NAT network.

If you are not using these adapters, you may wish to remove them (users on Windows hosts can choose to disable the adapters instead of removing them). The presence of these adapters has a slight performance cost, because broadcast packets must go to the extra adapters. On Windows networks, browsing your network may be slower than usual. And in some cases, these adapters interact with the host computer's networking configuration in undesirable ways.

Disabling a Host Virtual Adapter on a Windows Host

Use the Virtual Network Editor to disable any unwanted adapters.

1. Choose **Host** > **Virtual Network Settings** > **Host Virtual Adapters**.



2. Select the adapter you want to disable.
3. Click **Disable adapter**.
4. Click **OK**.

Enabling a Disabled Host Virtual Adapter on a Windows Host

Follow these steps to enable a host virtual adapter that is currently disabled on a Windows host.

1. Choose **Host** > **Virtual Network Settings** > **Host Virtual Adapters**.
2. Select the disabled adapter you want to enable.
3. Click **Enable adapter**.
4. Click **OK**.

Adding a Host Virtual Adapter to a Windows Host

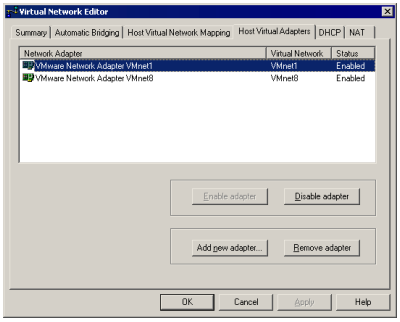
Follow these steps to add a host virtual adapter to a Windows host.

1. Choose **Host** > **Virtual Network Settings** > **Host Virtual Adapters**.
2. Click **Add new adapter**.
3. Choose the virtual network on which you want to use the adapter and click **OK**.
4. Click **Apply**.
5. Click **OK** to close the Virtual Network Editor.

Removing a Host Virtual Adapter from a Windows Host

Follow these steps to remove a host virtual adapter from a Windows host.

1. Choose **Host** > **Virtual Network Settings** > **Host Virtual Adapters**.



2. Select the adapter you want to remove, then click **Remove adapter**.
3. Click **OK**.

Removing a Host Virtual Adapter from a Linux Host

Follow these steps to remove a host virtual adapter from a Linux host.

1. Become root and run the GSX Server configuration program.

   ```
   su -
   vmware-config.pl
   ```

   Caution: In order to configure GSX Server correctly, the `vmware-config.pl` configuration program requires all virtual machines to be shut down. The program shuts down any running virtual machines automatically.

2. Watch for the following question:

   ```
   Do you want networking for your Virtual Machines? (yes/no/
   help) [yes]
   ```

Answer `yes` if you still want to use any networking in your virtual machines, then continue to the next question.

Otherwise, answer `no` to remove all networking.

3. If you answer `yes`, the script prompts you to select the wizard or editor to edit your network configuration. Select `editor`. This is the only way to delete virtual network adapters without removing all of them.

```
Would you prefer to modify your existing networking
configuration using the wizard or the editor? (wizard/
editor/help) [wizard] editor
```

4. You see a list of virtual networks that have been configured. Select the network corresponding to the adapter you wish to disable.

```
The following virtual networks have been defined:

. vmnet0 is bridged to eth0
. vmnet1 is a host-only network on subnet 172.16.155.0.
. vmnet8 is NAT network on a private subnet 172.16.107.0.

Which virtual network do you wish to configure? (0-99) 1
```

5. You may be prompted to keep this virtual network. If you are sure you want to remove it, answer `yes` to the question.

```
The network vmnet1 has been reserved for a host-only
network. You may change it, but it is highly recommended
that you use it as a host-only network. Are you sure you
want to modify it? (yes/no) [no] yes
```

6. When prompted about the type of virtual network, select `none` to remove the virtual network.

```
What type of virtual network do you wish to set vmnet1?
(bridged,hostonly,nat,none) [hostonly] none
```

Advanced Networking Topics

The following sections describe advanced networking topics:

- Selecting IP Addresses on a Host-Only Network or NAT Configuration on page 250
- Avoiding IP Packet Leakage in a Host-Only Network on page 253
- Maintaining and Changing the MAC Address of a Virtual Machine on page 255
- Controlling Routing Information for a Host-Only Network on a Linux Host on page 256
- Other Potential Issues with Host-Only Networking on a Linux Host on page 257
- Setting Up a Second Bridged Network Interface on a Linux Host on page 258
- Configuring Bridged Networking When Using Teamed Network Interface Cards on Your Host on page 259
- Setting Up Two Separate Host-Only Networks on page 261
- Routing Between Two Host-Only Networks on page 264
- Using Virtual Ethernet Adapters in Promiscuous Mode on a Linux Host

Selecting IP Addresses on a Host-Only Network or NAT Configuration

A host-only network uses a private virtual network. The host and all virtual machines configured for host-only networking are connected to the network through a virtual switch. Typically all the parties on this private network use the TCP/IP protocol suite, although other communication protocols may be used.

A network address translation (NAT) configuration also sets up a private network, which must be a TCP/IP network. The virtual machines configured for NAT are connected to that network through a virtual switch. The host computer is also connected to the private network used for NAT via a host virtual adapter.

Each virtual machine and the host must be assigned addresses on the private network. This is typically done using the DHCP server that comes with GSX Server. Note that this server does not service virtual (or physical) machines residing on bridged networks.

Addresses can also be assigned statically from a pool of addresses that are not assigned by the DHCP server.

When host-only networking is enabled at the time GSX Server is installed, the network number to use for the virtual network is automatically selected as an unused private IP network number. To find out what network is used on a Windows host, choose **Host** > **Virtual Network Settings** and check the subnet number associated with the virtual network. On a Linux host, run `ifconfig` in a terminal.

A NAT configuration also uses an unused private network automatically selected when you install GSX Server. To find out what network is used on a Windows host, choose **Host** > **Virtual Network Settings** and check the subnet number associated with the virtual network. On a Linux host, run `ifconfig` in a terminal.

Using DHCP to assign IP addresses is simpler and more automatic than statically assigning them. Most Windows operating systems, for example, come preconfigured to use DHCP at boot time, so Windows virtual machines can connect to the network the first time they are booted, without additional configuration. If you want your virtual machines to communicate with each other using names instead of IP addresses, however, you must set up a naming convention, a name server on the private network, or both. In that case it may be simpler to use static IP addresses.

In general, if you have virtual machines you intend to use frequently or for extended periods of time, it is probably most convenient to assign them static IP addresses or to configure the VMware DHCP server to always assign the same IP address to each of these virtual machines.

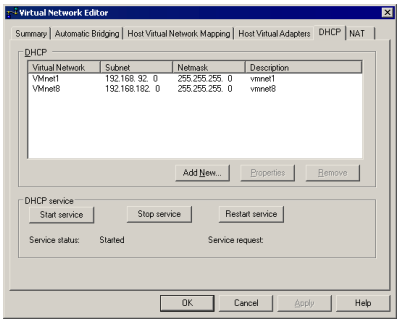
Configuring the DHCP Server on a Linux Host

On a Linux host, you configure the host-only DHCP server by editing the DHCP configuration file for VMnet1 (`/etc/vmware/vmnet1/dhcp/dhcp.conf`). To configure the DHCP server for the NAT network, edit the configuration file for VMnet8 (`/etc/vmware/vmnet8/dhcp/dhcp.conf`).

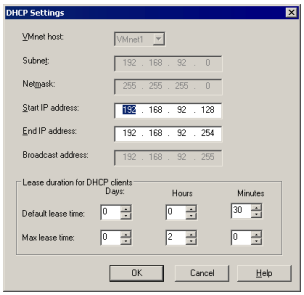
Editing the DHCP server configuration file requires information that is best obtained directly from the DHCP server documentation. Consult the manual pages `dhcpd(8)` and `dhcpd.conf(8)`.

Configuring the DHCP Server on a Windows Host

On a Windows host, you configure the DHCP server using the Virtual Network Editor (**Host** > **Virtual Network Settings** > **DHCP**).



Select the virtual network for which you want to change settings and click **Properties**.



Make the desired changes, then click **OK**.

Choosing the Method for Assigning IP Addresses

For virtual machines that you do not expect to keep for long, use DHCP and let it allocate an IP address.

For each host-only or NAT network, the available IP addresses are split up using the conventions shown in the tables below, where <net> is the network number assigned to your host-only or NAT network. GSX Server always uses a Class C address for host-only and NAT networks.

Address Use on a Host-Only Network

Range	Address use	Example
<net>.1	Host machine	192.168.0.1
<net>.2–<net>.127	Static addresses	192.168.0.2–192.168.0.127
<net>.128–<net>.253	DHCP-assigned	192.168.0.128–192.168.0.253
<net>.254	DHCP server	192.168.0.254
<net>.255	Broadcasting	192.168.0.255

Address Use on a NAT Network

Range	Address use	Example
<net>.1	Host machine	192.168.0.1
<net>.2	NAT device	192.168.0.2
<net>.3–<net>.127	Static addresses	192.168.0.3–192.168.0.127
<net>.128–<net>.253	DHCP-assigned	192.168.0.128–192.168.0.253
<net>.254	DHCP server	192.168.0.254
<net>.255	Broadcasting	192.168.0.255

Avoiding IP Packet Leakage in a Host-Only Network

By design, each host-only network should be confined to the host machine on which it is set up. That is, no packets sent by virtual machines on this network should leak out to a physical network attached to the host. Packet leakage can occur only if a machine actively forwards packets. It is possible for the host machine or any virtual machine running on the host-only network to be configured in a way that permits packet leakage.

Windows Hosts

Systems using server versions of Windows 2000 are capable of forwarding IP packets that are not addressed to them. By default, however, these systems come with IP packet forwarding disabled.

If you find packets leaking out of a host-only network on a Windows 2000 host computer, check to see if forwarding has been enabled on the host machine. If it is enabled, disable it.

Choose Start > Programs > Administrative Tools > Routing and Remote Access. An icon on the left is labeled with the host name. If a green dot appears over the icon, IP

forwarding is turned on. To turn it off, right-click the icon and disable **Routing and Remote Access**. A red dot appears, indicating that IP forwarding is disabled.

Linux Hosts

If you find packets leaking out of a host-only network on a Linux host computer, check to see if forwarding has mistakenly been enabled on the host machine. If it is enabled, disable it.

For many Linux systems, disable forwarding by writing a 0 (zero) to the special file `/proc/sys/net/ipv4/ip_forward`. As root, enter this command:

```
echo 0 > /proc/sys/net/ipv4/ip_forward
```

Other Linux systems have a system configuration option that you can set. The method depends on your Linux distribution. You may use a control panel, specify a setting at the time you compile your kernel or possibly enter a specification when you boot your system. Consult your operating system documentation for details on the method to use with your particular distribution.

Using Filtering

If the host computer has multiple network adapters, it may be intentionally configured to do IP forwarding. In this case, you do not want to disable forwarding. To avoid packet leakage, you must enable a packet filtering facility and specify that packets from the host-only network should not be sent outside the host computer. Consult your operating system documentation for details on how to configure packet filtering.

Leaks from a Virtual Machine

Virtual machines may leak packets, as well. For example, if you use dial-up networking support in a virtual machine and packet forwarding is enabled, host-only network traffic may leak out through the dial-up connection.

To prevent the leakage, be sure packet forwarding is disabled in your guest operating system.

Maintaining and Changing the MAC Address of a Virtual Machine

When a virtual machine is powered on, GSX Server automatically assigns each of its virtual network adapters an Ethernet MAC address. MAC stands for media access control. A MAC address is the unique address assigned to each Ethernet device.

The software guarantees that virtual machines are assigned unique MAC addresses within a given host system. In most cases, the virtual machine is assigned the same MAC address every time it is powered on, so long as the virtual machine is not moved (the path and filename for the virtual machine's configuration file must remain the same) and no changes are made to certain settings in that file.

In addition, GSX Server does its best, but cannot guarantee, to automatically assign unique MAC addresses for virtual machines running on multiple host systems.

Avoiding MAC Address Changes

To avoid changes in the MAC address automatically assigned to a virtual machine, you must not move the virtual machine's configuration file. Moving it to a different host computer or even moving it to a different location on the same host computer changes the MAC address.

You also need to be sure not to change certain settings in the virtual machine's configuration files. If you never edit the configuration file by hand and do not remove the virtual Ethernet adapter, these settings remain untouched. If you do edit the configuration file by hand, be sure not to remove or change the following options:

```
ethernet[n].generatedAddress
ethernet[n].addressType
ethernet[n].generatedAddressOffset
uuid.location
uuid.bios
ethernet[n].present
```

In these options, `[n]` is the number of the virtual Ethernet adapter, for example `ethernet0`.

Note: To preserve a virtual Ethernet adapter's MAC address, you also must be careful not to remove it. If you remove the adapter, then recreate it, it may receive a different MAC address.

Manually Assigning a MAC Address

If you want to guarantee that the same MAC address is assigned to a given virtual machine every time, even if the virtual machine is moved, or if you want to guarantee

a unique MAC address for each virtual machine within a networked environment, you can assign the address manually instead of letting GSX Server assign it automatically.

To assign the same, unique MAC address to any virtual machine manually, use a text editor to remove three lines from the configuration file and add one line. The configuration file has a `.vmx` extension at the end of the filename. On a Linux host, a virtual machine created with an earlier VMware product may have a configuration file with a `.cfg` extension.

Remove the three lines that begin with the following:

```
ethernet[n].generatedAddress
ethernet[n].addressType
ethernet[n].generatedAddressOffset
```

In these options, `[n]` is the number of the virtual Ethernet adapter — for example `ethernet0`.

Add the following line to the configuration file :

```
ethernet0.address = 00:50:56:XX:YY:ZZ
```

In this line, `XX` must be a valid hexadecimal number between `00h` and `3Fh`, and `YY` and `ZZ` must be valid hexadecimal numbers between `00h` and `FFh`. Because GSX Server virtual machines do not support arbitrary MAC addresses, you must use the above format.

So long as you choose a value for `XX:YY:ZZ` that is unique among your hard-coded addresses (where `XX` is a valid hexadecimal number between `00h` and `3Fh`, and `YY` and `ZZ` are valid hexadecimal numbers between `00h` and `FFh`), conflicts between the automatically assigned MAC addresses and the manually assigned ones should never occur.

Controlling Routing Information for a Host-Only Network on a Linux Host

A host-only network is a full-fledged network. It has a network interface associated with it (VMnet1) that is marked "up" at the time the host operating system is booted. Consequently, routing server processes that operate on the host operating system, such as `routed` and `gated`, automatically discover the network and propagate information on how to reach it unless you explicitly configure them not to do so.

If either of these processes is being run only to receive routing information, the easiest solution is to run the process with a `-q` option so that it does not supply routing information, only receives it.

If, however, the processes are running because they supply routing information, then you need to configure them so they do not advertise routes to the host-only network.

Unfortunately, the version of `routed` that comes with many distributions of Linux has no support for specifying that an interface should not be advertised. Consult the `routed(8)` manual page for your system in case you have a more contemporary version of the software.

The `gated` process requires some configuration. You need to explicitly exclude the VMnet1 interface from any protocol activity. If you need to run virtual machines on a host-only network on a multihomed system where `gated` is used and have problems doing so, please contact VMware technical support by submitting a support request at *www.vmware.com/requestsupport*.

Other Potential Issues with Host-Only Networking on a Linux Host

The following are common issues you may encounter when you are configuring a host-only network.

DHCPD on the Linux Host Does Not Work After GSX Server Installation

If you were running the DHCP server program `dhcpd` on your machine before installing GSX Server, it probably was configured to respond to DHCP requests from clients on any network interface present on the machine. When host-only networking is configured, an additional network interface, VMnet1, is marked "up" and available for use, and `dhcpd` may notice this.

In such cases, some `dhcpd` implementations abort if their configuration files do not include a subnet specification for the interface — even if `dhcpd` is not supposed to respond to messages that arrive through the interface.

The best solution to this problem is to add a line in the following format to the `dhcpd` configuration file:

```
subnet <net>.0 netmask 255.255.255.0 {}
```

`<net>` is the network number assigned to your host-only network — for example, 192.168.0. This configuration file entry informs `dhcpd` about the host-only network and tells it explicitly not to respond to any DHCP requests it sees coming from it.

An alternative solution is to explicitly state the set of network interfaces that you want `dhcpd` to listen to each time you start the program. For example, if your machine has one Ethernet interface, `eth0`, then each time you start `dhcpd`, list it on the command line:

```
dhcpd eth0
```

This keeps `dhcpd` from probing for all available network interfaces.

If the above solutions do not work for your DHCP server program, then it likely is old. You can try upgrading to a more current version such as the DHCP software available from the ISC Web site at *www.isc.org*.

DHCP and Dynamic Domain Name Service (DDNS)
DHCP can be used to hand out IP addresses as well as other information, such as the identity of a host running a name server and the nearest router or gateway. The DHCP server in GSX Server does not provide a means to dynamically establish a relationship between the IP address it assigns and a client's name (that is, to update a DNS server using DDNS).

If you want to use names to communicate with other virtual machines, you must either edit the DHCP configuration file for VMnet1 (`/etc/vmware/vmnet1.conf`) or use IP addresses that are statically bound to a host name. Editing the DHCP server configuration file requires information that is best obtained directly from the DHCP server documentation. Consult the manual pages `dhcpd(8)` and `dhcpd.conf(8)`.

Setting Up a Second Bridged Network Interface on a Linux Host

If you have two Ethernet adapters installed on your host computer, connected to two different networks, you may want your virtual machines on that host computer to bridge to both Ethernet adapters so the virtual machines can access either or both physical networks.

When you install GSX Server on a host computer with multiple Ethernet adapters, you have the option of configuring more than one bridged network. You can also configure additional bridged networks at any time by rerunning `vmware-config.pl`.

1. On the host computer, become root (`su -`) and run the GSX Server configuration program.

   ```
   vmware-config.pl
   ```

   Caution: In order to configure GSX Server correctly, the `vmware-config.pl` configuration program requires all virtual machines to be shut down. The program shuts down any running virtual machines automatically.

2. If you have more than one physical Ethernet adapter, one of the prompts you see is similar to this:

   ```
   The following bridged networks have been defined:
   . vmnet0 is bridged to eth0
   ```

```
Do you wish to configure another bridged network? (yes/no)
[no]
```

Answer `yes`.

3. If you have additional physical Ethernet adapters not yet connected to a bridged network, the prompt is repeated, showing information about all currently configured bridged networks.
4. When you have set up all the bridged networks you want, type `no`.

Configuring Bridged Networking When Using Teamed Network Interface Cards on Your Host

Network adapter teaming (where two or more network interface cards work together as one and appear as a single, separate device) provides a GSX Server host and the virtual machines running on it with a level of network hardware fault tolerance. Should one physical network adapter fail, network traffic for the host and virtual machines can continue using the remaining network adapters in the team.

Another method for providing fault tolerance is by making sure that automatic bridging is enabled. This feature is available on Windows hosts only and is enabled by default. For more information, see Configuring Bridged Networking Options on a Windows Host on page 242. This method is more limited than using NIC teaming, as it does not allow for load balancing, switch fault tolerance, fault tolerance to any necessary services running on the host or the ability to specify an adapter as the primary or secondary adapter.

Certain NIC teaming modes provide load balancing and are discussed below.

If your GSX Server host is configured to use teamed network interface cards, and you use bridged networking with your virtual machines, you need to adjust your network settings. You do this by binding the VMware Bridge Protocol to the teamed NIC and unbinding it from each individual, physical NIC on the host. See Setting Up the Windows Host on page 260.

Before you start using teamed NICs to network your virtual machines, you should have a good understanding of how network teaming works in your host environment.

Support for Network Adapter Teaming

VMware supports teamed NICs on Windows hosts with enterprise class network adapters that can be configured for NIC teaming. If there is a specific teamed networking mode (such as 802.3ad Dynamic or 802.3ad-Draft Static mode) you want to use, then you should use adapters that support that mode.

VMware has not tested and does not support network adapter teams with GSX Server on Linux hosts.

GSX Server supports teamed Broadcom-based network adapters when used with Broadcom teaming software in the following modes:

- Generic Trunking (FEC/GEC/802.3ad-Draft Static)
- Link Aggregation (802.3ad)
- Smart Load Balance and Fail Over

GSX Server supports teamed Intel-based network adapters when used with Intel PROSet version 6.4 or higher (32-bit hosts) or PROSet version 10.0 or higher (64-bit hosts) in the following modes:

- Adapter Fault Tolerance
- Adaptive Load Balancing
- Static Link Aggregation (64-bit hosts)
- FEC/802.3ad Static Link Aggregation (32-bit hosts)
- GEC/802.3ad Static Link Aggregation (32-bit hosts)
- IEEE 802.3ad Dynamic Link Aggregation

Note: Express Teaming mode is not supported when you are teaming Intel-based network adapters.

Setting Up the Windows Host

When using GSX Server on a Windows host with teamed network adapters and bridged networking, the VMware Bridge Protocol must be bound to the teamed network adapter and unbound from the individual physical network adapters. Complete the following steps.

1. Open the Windows Control Panel, then open Network Connections (on a Windows Server 2003 host) or open Network and Dial-up Connections (on a Windows 2000 host).
2. To bind the VMware Bridge Protocol to the teamed NIC, right-click the teamed NIC device and choose **Properties**. Check **VMware Bridge Protocol**, then click **OK** to close the property sheet.
3. To unbind the VMware Bridge Protocol from each physical NIC that is being used for bridged networking, right-click the NIC device and choose **Properties**. Clear the **VMware Bridge Protocol** check box, then click **OK** to close the property sheet.

Alternately, you can use the Virtual Network Editor to either map the teamed NIC to VMnet0 or exclude the physical adapters from any automatic bridging by GSX Server. For information, see Configuring Bridged Networking Options on a Windows Host on page 242.

Changing the Teamed Networking Mode

If you change the teamed networking mode, you must delete the original NIC team on the host and create a new team. Do not modify a virtual machine's NIC teaming settings.

Caution: Before you delete the original team, power off or suspend all virtual machines on the host to prevent the teaming software from locking up.

Setting Up Two Separate Host-Only Networks

For some configurations, you may need to set up more than one host-only network on the same host computer.

You may, for example, want to have two virtual machines connected to one host-only network, and at the same time have other virtual machines connected to another host-only network so the network traffic on each network is isolated.

Or you may want to test routing between two virtual networks. Or test a virtual machine with multiple network interface cards — without using any physical Ethernet adapters.

On Windows hosts, the first host-only network is set up automatically when you install GSX Server.

On Linux hosts, the first host-only network is set up when you run the `vmware-config.pl` program after you install GSX Server (provided you agree to install host-only networking). If you did not agree to use host-only networking, you need to run the script again to set up host-only networking.

To set up the second host-only network, follow the steps outlined below for your host operating system.

Setting Up the Second Host-Only Interface on a Windows Host

Follow these steps to set up the second host-only interface on a Windows host.

1. Choose **Host** > **Virtual Network Settings** > **Host Virtual Adapters**.
2. Click **Add new adapter**.
3. Choose the virtual network on which you want to use the adapter and click **OK**.
4. Click **Apply**.

5. Click **OK** to close the Virtual Network Editor.

Setting Up the Second Host-Only Interface on a Linux Host

1. As root (`su -`), run the GSX Server configuration program.
   `vmware-config.pl`

   Caution: In order to configure GSX Server correctly, the `vmware-config.pl` configuration program requires all virtual machines to be shut down. The program shuts down any running virtual machines automatically.

2. Use the wizard to modify your configuration. After asking about a NAT network, the program asks:

   `Do you want to be able to use host-only networking in your virtual machines?`

   Answer `yes`.

   The wizard reports on host-only networks that you have already set up on the host or, if no host-only network is present, configures the first one.

3. The wizard asks:

   `Do you wish to configure another host-only network?`

   Answer `yes`.

   Repeat this step until you have as many host-only networks as you want. Then answer `no`.

4. Complete the remaining steps in the wizard. When the wizard is finished, it restarts all services used by GSX Server.

5. Run `ifconfig`. You should see at least four network interfaces — `eth0`, `lo`, `vmnet1` and `vmnet2`. If the VMnet interfaces do not show up immediately, wait for a minute, then run the command again. These four interfaces should have different IP address on separate subnets.

Configuring the Virtual Machines

Now you have two host-only interfaces (VMnet1 and VMnet2). You are ready to set up your virtual machines for one of the following configurations:

1. The virtual machine is configured with one virtual Ethernet adapter, and that virtual adapter is connected to the default host-only interface (VMnet 1).
2. The virtual machine is configured with one virtual Ethernet adapter, and that virtual adapter is connected to the newly created host-only interface (VMnet2).

3. The virtual machine is configured with two virtual Ethernet adapters. One virtual adapter is connected to the default host-only interface (VMnet1) and the other virtual adapter is connected to the newly created host-only interface (VMnet2).

Configuration 1 — Connect to the Default Host-Only Interface

1. Create the virtual machine or use an existing virtual machine.
2. Launch a VMware Virtual Machine Console and open the virtual machine.
3. Edit the configuration using the virtual machine settings editor (**VM** > **Settings**).

   Select **NIC**, select **Custom**, then choose **VMnet1 (Host-only)** (on a Windows host) or **/dev/vmnet1** (on a Linux host) from the drop-down list on the right.

   If no network adapter is shown in the list of devices, click **Add**, then use the Add Hardware Wizard to add an adapter.

Configuration 2 — Connect to the Newly Created Host-Only Interface

1. Create the virtual machine or use an existing virtual machine.
2. Launch a VMware Virtual Machine Console and open the virtual machine.
3. Edit the configuration using the virtual machine settings editor (**VM** > **Settings**).

   Select **NIC**, select **Custom**, then choose **VMnet 2 (Host-only)** (on a Windows host) or **/dev/vmnet2** (on a Linux host) from the drop-down list on the right.

   If no network adapter is shown in the list of devices, click **Add**, then use the Add Hardware Wizard to add an adapter.

Configuration 3 — Connect to Two Host-Only Interfaces

1. Create the virtual machine or use an existing virtual machine.
2. Launch VMware GSX Server and open the virtual machine.
3. Edit the configuration using the virtual machine settings editor (**VM** > **Settings**).

   Select the first network adapter in the list of devices, select **Custom**, then choose **VMnet1 (Host-only)** (on a Windows host) or **/dev/vmnet1** (on a Linux host) from the drop-down list on the right. Select the second network adapter in the list of devices, select **Custom**, then choose **VMnet 2 (Host-only)** (on a Windows host) or **/dev/vmnet2** (on a Linux host) from the drop-down list on the right.

   If you need to add one or more network adapters, click **Add**, then use the Add Hardware Wizard to add an adapter.

At this point you can power on the virtual machine and install your guest operating system. In configurations 1 and 2 you see one AMD PCNet Family Adapter. In configuration 3 you see two AMD PCNet Family Adapters within the guest. Configure

the Ethernet adapters as you would physical adapters on a physical computer, giving each adapter an IP address on the appropriate VMnet subnet.

On Windows hosts, you can open a command prompt and run `ipconfig /all` to see what IP addresses each host-only network is using.

On Linux hosts, you can open a terminal and run `ifconfig` to see what IP addresses each host-only network is using.

Routing Between Two Host-Only Networks

If you are setting up a complex test network using virtual machines, you may want to have two independent host-only networks with a router between them.

There are two basic approaches. In one, the router software runs on the host computer. In the other, the router software runs in its own virtual machine. In both cases, you need two host-only interfaces.

The examples described here outline the simplest case, with one virtual machine on each of the host-only networks. For more complex configurations, you can add more virtual machines and host-only networks as appropriate.

Setting Up the First Host-Only Interface

On Windows hosts, the first host-only network is set up automatically when you install GSX Server.

On Linux hosts, the first host-only network is set up when you run the `vmware-config.pl` program after you install GSX Server, provided you agree to install host-only networking. If you did not agree to use host-only networking, you need to run the script again to set up host-only networking.

Setting Up the Second Host-Only Interface on a Windows Host

Follow these steps to set up the second host-only interface on a Windows host.

1. Go to **Host** > **Virtual Network Settings** > **Host Virtual Adapters**.
2. Click **Add new adapter**.
3. Choose the virtual network on which you want to use the adapter and click **OK**.
4. Click **Apply**.
5. Click **OK** to close the Virtual Network Editor.

Setting Up the Second Host-Only Interface on a Linux Host

1. As root (`su -`), run the GSX Server configuration program.

   ```
   vmware-config.pl
   ```

Caution: In order to configure GSX Server correctly, the `vmware-config.pl` configuration program requires all virtual machines to be shut down. The program shuts down any running virtual machines automatically.

2. Use the wizard to modify your configuration. After asking about a NAT network, the program asks:

   ```
   Do you want to be able to use host-only networking in your
   virtual machines?
   ```

   Answer `yes`.

   The wizard reports on host-only networks that you have already set up on the host or, if none is present, configures the first host-only network.

3. The wizard asks:

   ```
   Do you wish to configure another host-only network?
   ```

   Answer `yes`.

   Repeat this step until you have as many host-only networks as you want. Then answer `no`.

4. Complete the wizard. When it is finished, it restarts all services used by GSX Server.

5. Run `ifconfig`. You should see at least four network interfaces — `eth0`, `lo`, `vmnet1` and `vmnet2`. If the VMnet interfaces do not show up immediately, wait for a minute, then run the command again. These four interfaces should have different IP address on separate subnets.

Setting Up the Virtual Machines

Now you have two host-only network adapters on the host computer. Each is connected to its own virtual switch (VMnet1 and VMnet2). You are ready to create and configure your virtual machines and connect them to the appropriate virtual switches.

Virtual Machine 1 — Connected to the Default Host-Only Interface

1. Create the virtual machine or use an existing virtual machine.
2. Launch a VMware Virtual Machine Console and open the virtual machine.
3. Edit the configuration using the virtual machine settings editor (**VM** > **Settings**).

   Select **NIC**, select **Custom**, then choose **VMnet1 (Host-only)** (on a Windows host) or **/dev/vmnet1** (on a Linux host) from the drop-down list on the right.

   If no network adapter is shown in the list of devices, click **Add**, then use the Add Hardware Wizard to add an adapter.

Virtual Machine 2 — Connected to the Newly Created Host-Only Interface

1. Create the virtual machine or use an existing virtual machine.
2. Launch a VMware Virtual Machine Console and open the virtual machine.
3. Edit the configuration using the virtual machine settings editor (**VM** > **Settings**).

   Select **NIC**, select **Custom**, then choose **VMnet2 (Host-only)** (on a Windows host) or **/dev/vmnet2** (on a Linux host) from the drop-down list on the right.

   If no network adapter is shown in the list of devices, click **Add**, then use the Add Hardware Wizard to add an adapter.

If you plan to run the router software on your host computer, you can skip the next section.

Virtual Machine 3 — Connected to Both Host-Only Interfaces

If you plan to run the router software on a virtual machine, set up a third virtual machine for that purpose.

1. Create the virtual machine or use an existing virtual machine.
2. Launch a VMware Virtual Machine Console and open the virtual machine.
3. Edit the configuration using the virtual machine settings editor (**VM** > **Settings**).

   Select the first network adapter in the list of devices, select **Custom**, then choose **VMnet1 (Host-only)** (on a Windows host) or **/dev/vmnet1** (on a Linux host) from the drop-down list on the right. Select the second network adapter in the list of devices, then select **Custom**, choose **VMnet 2 (Host-only)** (on a Windows host) or **/dev/vmnet2** (on a Linux host) from the drop-down list on the right.

   If you need to add one or more network adapters, click **Add**, then use the Add Hardware Wizard to add an adapter.

Now you need to configure the networking components on the host and in the virtual machines. The recommended approach uses static IP addresses for all the virtual machines.

1. Stop the VMnet DHCP server service.

   Windows host: Choose **Host** > **Virtual Network Settings** > **DHCP** and click **Stop service**.

   Linux host: Stop the `vmnet-dhcpd` service.

   ```
   killall -TERM vmnet-dhcpd
   ```

2. Install guest operating systems in each of the virtual machines.

3. Install the router software — on the host computer or in the third virtual machine, depending on the approach you are using.
4. Configure networking in the first two virtual machines to use addresses on the appropriate host-only network.

   On Windows hosts, you can open a command prompt and run `ipconfig /all` to see what IP addresses each host-only network is using.

   On Linux hosts, you can open a terminal and run `ifconfig` to see what IP addresses each host-only network is using.
5. If you are running the router on the host computer, assign default router addresses based on the addresses of the host-only adapters on the host computer. In the first virtual machine's networking configuration, the default router address should be the IP address for the host-only adapter connected to VMnet1. In the second virtual machine's networking configuration, the default router address should be the IP address for the host-only adapter connected to VMnet2.

   If you are running the router software on the third virtual machine, set the default router addresses in the first two virtual machines based on those used by the third virtual machine. In the first virtual machine's networking configuration, the default router address should be the IP address for the third virtual machine's Ethernet adapter connected to VMnet1. In the second virtual machine's networking configuration, the default router address should be the IP address for the third virtual machine's Ethernet adapter connected to VMnet2.

At this point you should be able to ping the router machine from virtual machines one and two. And if the router software is set up correctly, you should be able to communicate between the first and second virtual machines.

Using Virtual Ethernet Adapters in Promiscuous Mode on a Linux Host

GSX Server does not allow the virtual Ethernet adapter to go into promiscuous mode unless the user running GSX Server has permission to make that setting. This follows the standard Linux practice that only root can put a network interface into promiscuous mode.

When you install and configure GSX Server, you must run the installation as root. GSX Server creates the VMnet devices with root ownership and root group ownership, which means that only root has read and write permissions to the devices.

To set the virtual machine's Ethernet adapter to promiscuous mode, you must launch GSX Server as root because you must have read and write access to the VMnet device. For example, if you are using bridged networking, you must have access to `/dev/vmnet0`.

To grant selected other users read and write access to the VMnet device, you can create a new group, add the appropriate users to the group and grant that group read and write access to the appropriate device. You must make these changes on the host operating system as root (`su -`). For example, you can enter the following commands:

```
chgrp <newgroup> /dev/vmnet0
chmod g+rw /dev/vmnet0
```

`<newgroup>` is the group that should have the ability to set `vmnet0` to promiscuous mode.

If you want all users to be able to set the virtual Ethernet Adapter (`/dev/vmnet0` in our example) to promiscuous mode, you can simply run the following command on the host operating system as root:

```
chmod a+rw /dev/vmnet0
```

Understanding NAT

Network address translation — or NAT — provides a simple way for virtual machines to use most client applications over almost any type of network connection available to the host. The only requirement for NAT is that the network connection must support TCP/IP.

NAT is useful when you have a limited supply of IP addresses or are connected to the network through a non-Ethernet network adapter. NAT works by translating addresses of virtual machines in a private VMnet network to that of the host machine. When a virtual machine sends a request to access a network resource, it appears to the network resource as if the request came from the host machine.

NAT uses the host's own network resources to connect to the external network. Thus, any TCP/IP network resource to which the host has access should be available through the NAT connection.

The chief advantage of NAT is that it provides a transparent, easy to configure way for virtual machines to gain access to network resources.

The following sections provide more information about NAT:

- Using NAT on page 269
- The Host Computer and the NAT Network on page 270
- DHCP on the NAT Network on page 270
- DNS on the NAT Network on page 270
- External Access from the NAT Network on page 271
- Advanced NAT Configuration on page 272
- Custom NAT and DHCP Configuration on a Windows Host on page 275
- Considerations for Using NAT on page 276
- Using NAT with NetLogon on page 277
- Sample Linux vmnetnat.conf File on page 278

Using NAT

The NAT device is connected to the VMnet8 virtual switch. Virtual machines connected to the NAT network also use the VMnet8 virtual switch.

The NAT device waits for packets coming from virtual machines on the VMnet8 virtual network. When a packet arrives, the NAT device translates the address of the virtual machine to that of the host before forwarding the packet to the external network.

When data arrives from the external network for the virtual machine on the private network, the NAT device receives the data, replaces the network address with that of the virtual machine and forwards the data to the virtual machine on the virtual network. This translation occurs automatically and requires minimal configuration on the guest and the host.

The Host Computer and the NAT Network

The host computer has a host virtual adapter on the NAT network (identical to the host virtual adapter on the host-only network). This adapter allows the host and the virtual machines to communicate with each other for such purposes as file sharing. The NAT never forwards traffic from the host virtual adapter.

DHCP on the NAT Network

In order to make networking configuration easy, a DHCP server is automatically installed when you install GSX Server. Virtual machines running on the network with the NAT device can dynamically obtain their IP addresses by sending out DHCP requests. The DHCP server on the NAT network, which is also used in host-only networking configurations, dynamically allocates IP addresses in the range of <net>.128 through <net>.254, where <net> is the network number assigned to your NAT network. GSX Server always uses a Class C address for NAT networks. IP addresses <net>.3 through <net>.127 can be used for static IP addresses. IP address <net>.1 is reserved for the host adapter; <net>.2 is reserved for the NAT device.

In addition to the IP address, the DHCP server on the NAT network also sends out additional configuration information that enables the virtual machine to operate automatically. This information includes the default gateway and the DNS server. In the DHCP response, the NAT device instructs the virtual machine to use the IP address <net>.2 as the default gateway and DNS server. This causes all IP packets destined for the external network and DNS requests to be forwarded to the NAT device.

DNS on the NAT Network

The NAT device acts as a DNS server for the virtual machines on the NAT network. Actually, the NAT device is a DNS proxy and merely forwards DNS requests from the virtual machines to a DNS server that is known by the host. Responses come back to the NAT device, which then forwards them to the virtual machines.

If they get their configuration information from DHCP, the virtual machines on the NAT network automatically use the NAT device as the DNS server. However, the virtual machines can be statically configured to use another DNS server.

The virtual machines in the private NAT network are not, themselves, accessible via DNS. If you want the virtual machines running on the NAT network to access each other by DNS names, you must set up a private DNS server connected to the NAT network.

External Access from the NAT Network

In general, any protocol using TCP or UDP can be used automatically by a virtual machine on the NAT network so long as the virtual machine initiates the network connection. This is true for most client applications such as Web browsing, Telnet, passive-mode FTP and downloading streaming video. Additional protocol support has been built into the NAT device to allow FTP and ICMP echo (ping) to work completely transparently through the NAT.

On the external network to which the host is connected, any virtual machine on the NAT network appears to be the host itself, because its network traffic uses the host's IP address. It is able to send and receive data using TCP/IP to any machine that is accessible from the host.

Before any such communication can occur, the NAT device must set up a mapping between the virtual machine's address on the private NAT network and the host's network address on the external network.

When a virtual machine initiates a network connection with another network resource, this mapping is created automatically. The operation is perfectly transparent to the user of the virtual machine on the NAT network. No additional work needs to be done to let the virtual machine access the external network.

The same cannot be said for network connections that are initiated from the external network to a virtual machine on the NAT network.

When a machine on the external network attempts to initiate a connection with a virtual machine on the NAT network, it cannot reach the virtual machine because the NAT device does not forward the request. Network connections that are initiated from outside the NAT network are not transparent.

However, it is possible to configure port forwarding manually on the NAT device so network traffic destined for a certain port can still be forwarded automatically to a virtual machine on the NAT network. For details, see Advanced NAT Configuration on page 272.

File sharing of the type used by Windows operating systems and Samba is possible among computers on the NAT network — including virtual machines and the host computer. If you are using WINS servers on your network, a virtual machine using NAT networking can access shared files and folders on the host that are known by the

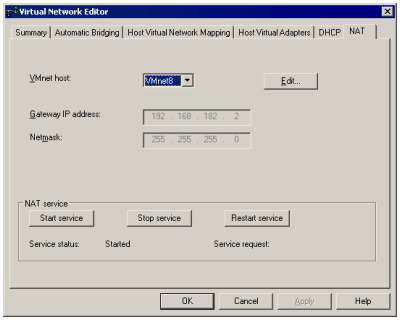
WINS server so long as those shared files and folders are in the same workgroup or domain.

Advanced NAT Configuration

Read the section that corresponds to your host operating system for information on configuring NAT for your virtual machines.

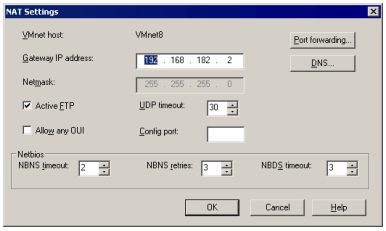
Windows Hosts

Configure the NAT device using the Virtual Network Editor (**Host** > **Virtual Network Settings** > **NAT**).



You can stop, restart and start the virtual NAT device by clicking the appropriate button. The **VMnet host** setting lets you choose which virtual network uses the NAT device. You can select **Disable** if you do not want to use NAT on any virtual network.

To edit NAT settings for a virtual network, choose it from the drop-down menu, then click **Edit**. The NAT Settings dialog box appears.



You can change any of the following NAT settings:

- **Port forwarding** lets you send incoming TCP or UDP requests to a specific virtual machine on the virtual network served by the NAT device. To set up and configure forwarded ports, click **Port forwarding**. A dialog box appears.

To add a new port for either TCP or UDP, click **Add**. If a port is already listed, you can change its settings. Select its name in the list, then click **Properties**. Or click **Remove** to remove the selected port.

When you click **Add**, another dialog box appears. In the **Host port** field, type the number of the incoming TCP or UDP port. For example, incoming HTTP requests are usually on port 80. In the first **Forwarding IP address** field, type the IP address of the virtual machine to which you want to forward the incoming requests. In the second field on that line, type the port number you want to use for those requests on that virtual machine. This may be the standard port, such as 80 for HTTP, or a nonstandard port if software running in the virtual machine is configured to accept requests on a nonstandard port. The **Description** field is optional. You may use it to identify the service being forwarded (for example, HTTP). When you have made these settings, click **OK**.

- You may specify DNS servers to be used by the virtual NAT device. To do so, click **DNS**. A dialog box appears. You can change the **Policy** for using multiple DNS servers if you prefer to use **Rotate** or **Burst** instead of the default setting of **Order**. To add a DNS server to the list, click **Add**. Another dialog box appears. Enter the DNS server's IP address in the **IP address** field. The **Description** field is optional. When you have made the desired settings, click **OK**.To change the settings for a server already in the list, select its entry in the DNS dialog box, then click **Properties**. To delete an entry, select the entry, then click **Remove**. When you have made the desired changes, click **OK**.
- You can change the IP address for the NAT device in the **Gateway IP address** field. If you need to change the Netmask, click the **...** button on the **Host Virtual Network Mapping** tab of the Virtual Network Editor and choose **Subnet**.
- If you want to allow only passive mode FTP over the NAT device, deselect **the Active FTP** check box.
- You can change the number of minutes to keep the UDP mapping for the NAT in the **UDP timeout** field.
- If you change the OUI (Organizationally Unique Identifier) portion of the MAC address for the virtual machine and subsequently cannot use NAT with the virtual machine, you should check the **Allow Any OUI** check box.
- In the **Config port** field, you can specify a port that can be used to access status information about the NAT. This option is used for troubleshooting purposes with VMware technical support only.
- You can change NetBIOS timeout and retry settings.

When you have made all the networking changes you want, click **OK**.

Linux Hosts

Use the NAT configuration file on the host to configure the NAT device. This file is `/etc/vmware/vmnet8/nat/nat.conf`.

The configuration file is divided into sections. Each section configures a part of the NAT device. Text surrounded by square brackets — such as `[host]` — marks the beginning of a section. In each section is a configuration parameter that can be set. The configuration parameters take the form `ip = 192.168.27.1/24`.

For an example of a NAT configuration file, see Sample Linux vmnetnat.conf File on page 278. The configuration file variables are described below.

The [host] Section

`ip`
The IP address that the NAT device should use. It can optionally be followed by a slash and the number of bits in the subnet.

`netmask`
The subnet mask to use for the NAT. DHCP addresses are allocated from this range of addresses.

`configport`
A port that can be used to access status information about the NAT.

`device`
The VMnet device to use. Linux devices are of the format `/dev/vmnet<x>`. VMnet8 is the default NAT device.

`activeFTP`
A flag that indicates if active FTP is to be allowed. Active FTP allows incoming connections to be opened by the remote FTP server. Turning this off means that only passive mode FTP works. Set the flag to `0` to turn active FTP off.

The [udp] Section

`timeout`
The number of minutes to keep the UDP mapping for the NAT.

The [incomingtcp] Section

Use this section to configure TCP port forwarding for NAT. You can assign a port number to an IP address and port number on a virtual machine.

The following line shows the format used in this section.

```
8887 = 192.168.27.128:21
```

This example creates a mapping from port 8887 on the host to the IP address 192.168.27.128 and port 21. When this mapping is set and an external machine connects to the host at port 8887, the network packets are automatically forwarded to port 21 (the standard port for FTP) on the virtual machine with IP address 192.168.27.128.

The [incomingudp] Section

Use this section to configure UDP port forwarding for NAT. You can assign a port number to an IP address and port number on a virtual machine.

The following line shows the format used in this section. It illustrates a way to forward X server traffic from the host port 6000 to the virtual machine's port 6001.

```
6000 = 192.168.27.128:6001
```

This example creates a mapping from port 6000 on the host to the IP address 192.168.27.128 and port 6001. When this mapping is set and an external machine connects to the host at port 6000, the network packets are automatically forwarded to port 6001 on the virtual machine with IP address 192.168.27.128.

Custom NAT and DHCP Configuration on a Windows Host

If you are an advanced user on a Windows host computer, you may wish to make custom configuration settings by editing the NAT and DHCP configuration files. If your host operating system is installed on the C drive, the configuration files for NAT and DHCP are in the following locations:

- **NAT:** `C:\Documents and Settings\All Users\Application Data\VMware\vmnetnat.conf`
- **DHCP:** `C:\Documents and Settings\All Users\Application Data\VMware\vmnetdhcp.conf`

Note: You can change many key NAT and DCHP settings using the Virtual Network Editor (**Host** > **Virtual Network Settings**). However, if you have made manual changes to the configuration files, some or all of those changes may be lost when you use the Virtual Network Editor. If you have made manual changes, you should make backup copies of the files before changing any settings in the Virtual Network Editor. After making changes in the Virtual Network Editor, you can copy your manual changes back into the appropriate configuration files.

Specifying Connections from Ports Below 1024

When a client machine makes a TCP or UDP connection to a server, the connection comes from a particular port on the client (the source port) and connects to a

particular port on the server (the destination port). For security reasons, some servers accept connections only from source ports below 1024.

If a virtual machine using NAT attempts to connect to a server that requires the client to use a source port below 1024, it is important that the NAT device forward the request from a port below 1024. You can specify this behavior in the `vmnetnat.conf` file.

This behavior is controlled by entries in sections headed `[privilegedUDP]` and `[privilegedTCP]`. You may need to add settings to or modify settings in either or both of these sections, depending on the kind of connection you need to make.

You can set two parameters, each of which appears on a separate line.

`autodetect = <n>`
The autodetect setting determines whether the VMware NAT device automatically attempts to map virtual machine source ports below 1024 to NAT source ports below 1024. A setting of `1` means true. A setting of `0` means false. On a Windows host, the default is `1` (true). On a Linux host, the default is `0` (false).

`port = <n>`
The port setting specifies a destination port (`<n>` is the port on the server that accepts the connection from the client). Whenever a virtual machine connects to the specified port on any server, the NAT device attempts to make the connection from a source port below 1024. You may include one or more port settings in the `[privilegedUDP]` or `[privilegedTCP]` section or in both sections, as required for the connections you need to make. Each port setting must be entered on a separate line.

Considerations for Using NAT

Because NAT requires that every packet sent and received from virtual machines be in the NAT network, there is an unavoidable performance penalty. Our experiments show that the penalty is minor for dial-up and DSL connections and performance is adequate for most GSX Server uses.

NAT is not perfectly transparent. It does not normally allow connections to be initiated from outside the network, although you can set up server connections by manually configuring the NAT device. The practical result is that some TCP and UDP protocols that require a connection be initiated from the server machine — some peer to peer applications, for example — do not work automatically, and some may not work at all.

A standard NAT configuration provides basic-level firewall protection because the NAT device can initiate connections from the private NAT network, but devices on the external network cannot normally initiate connections to the private NAT network.

Using NAT with NetLogon

When using NAT networking in a virtual machine with a Windows guest operating system running on a Windows host, you can use NetLogon to log on to a Windows domain from the virtual machine. You can then access file shares known by the WINS server in the domain.

To use NetLogon, you need to know how WINS servers and Windows domain controllers work. This section explains how to set up the virtual machine to use NetLogon. The setup process is similar to the way you set up a physical computer on one LAN that is using a domain controller on another LAN.

In order to log on to a Windows domain outside the virtual NAT network, the virtual machine needs access to a WINS server for that domain. There are two ways you can connect the virtual machine to a WINS server. You can connect to the WINS server provided by the DHCP server used on the NAT network, provided that the WINS server is already set up on the host. If you want to connect from the virtual machine to a WINS server not set up on the host, you can manually enter the IP address of the WINS server.

Using NAT to Connect to an Existing WINS Server Already Set Up on the Host

In order to use this method, a WINS server in the same workgroup or domain must be set up on the host. These steps use Windows 2000, Windows XP or Windows Server 2003 as a guide. The process is similar for Windows NT, Windows Me and Windows 9x guests.

1. In the virtual machine, right-click on **My Network Places** and select **Properties**.
2. In the Network Connections window, right-click the virtual network adapter and select **Properties**.
3. In the Properties dialog box, select **Internet Protocol (TCP/IP)**, then click **Properties**.
4. In the TCP/IP Properties dialog box, click **Advanced**.
5. Click the **WINS** tab, then under **NetBIOS setting**, select **Use NetBIOS setting from DHCP Server**.
6. Click **OK** twice, then click **Close**.

Manually Entering the IP Address of a WINS Server

Use this method to connect to a WINS server in the same workgroup or domain that is not already set up on the host.

1. In the virtual machine, right-click on **My Network Places** and select **Properties**.

2. In the Network Connections window, right-click the virtual network adapter and select **Properties**.
3. In the Properties dialog box, select **Internet Protocol (TCP/IP)**, then click **Properties**.
4. In the TCP/IP Properties dialog box, click **Advanced**.
5. Click the **WINS** tab, then click **Add**.
6. In the TCP/IP WINS Server dialog box, enter the IP address for the WINS server in the **WINS server** field, then click **OK**. The IP address of the WINS server appears in the **WINS addresses** list on the WINS tab.

   Repeat steps 5 and 6 for each WINS server to which you want to connect from this virtual machine.
7. Click **OK** twice, then click **Close**.

Now that the virtual machine has an IP address for a WINS server, you use NetLogon in the virtual machine to log on to a domain and access shares in that domain.

For example, if the WINS server covers a domain with a domain controller, it is possible to access that domain controller from the virtual machine and add the virtual machine to the domain. You need to know the user ID and password of the Administrator on the domain controller.

Note: Your access is limited to shares of virtual machines that are on the same NAT network or are bridged on the same domain.

Sample Linux vmnetnat.conf File

The following is a sample Linux `vmnetnat.conf` file.

```
# Linux NAT configuration file

[host]
# NAT gateway address
ip = 192.168.237.2/24
hostMAC = 00:50:56:C0:00:08

# enable configuration; disabled by default for security reasons
#configport = 33445

# VMnet device if not specified on command line
device = VMnet8

# Allow PORT/EPRT FTP commands (they need incoming TCP stream...)
activeFTP = 1
```

```
# Allows the source to have any OUI. Enable this if you change the OUI
# in the MAC address of your virtual machines.
#allowAnyOUI = 1

[udp]
# Timeout in seconds, 0 = no timeout, default = 60; real value might
# be up to 100% longer
timeout = 30

[incomingtcp]
# Use these with care - anyone can enter into your virtual machine through
# these...

# FTP (both active and passive FTP is always enabled)
#   ftp localhost 8887
#8887 = 192.168.27.128:21

# WEB (make sure that if you are using named webhosting, names point to
#   your host, not to guest... And if you are forwarding port other
#   than 80 make sure that your server copes with mismatched port
#   number in Host: header)
#   lynx http://localhost:8888
#8888 = 192.168.27.128:80

# SSH
#   ssh -p 8889 root@localhost
#8889 = 192.168.27.128:22

[incomingudp]
# UDP port forwarding example
#6000 = 192.168.27.128:6001
```

Using Samba for File Sharing on a Linux Host

On a Linux host computer, GSX Server can automatically install and configure a Samba server to act as a file server for Microsoft Windows guest operating systems.

You can then use Windows Explorer in the virtual machine to move and copy files between virtual machine and host — or between virtual machines on the same network — just as you would with files on physical computers that share a network connection.

The lightly modified Samba server installed by GSX Server runs over the GSX Server virtual Ethernet, and the Samba traffic between different operating systems is isolated from actual local area networks.

The source code differences for the changes (in `diff` format and based on Samba 2.0.6) are available from VMware. For more information, see *www.vmware.com/download/open_sources.html*.

If you already have Samba configured on your Linux host, the recommended approach is to modify that configuration so it includes the IP subnet used by the GSX Server virtual Ethernet adapter, VMnet1.

You can configure your existing Samba server to work with a host-only network. Note that all the shares you set up in Samba and in the guest operating system normally appear on the bridged network as well.

If you need to be sure the shares set up in the guest operating system are seen only on the host-only network, you may find it easiest to install and use the Samba server provided with GSX Server.

If you do not need any shares to appear on your bridged network, you can use your existing Samba server and set up the configuration file so it works only on the host-only network.

Samba configurations can be quite complex. This section provides several sample configuration files. If you need to go beyond the issues covered here, see the man page for the `smb.conf` file. To view this man page, type one of the following commands in a terminal window:

```
man smb.conf
```

or

```
man 5 smb.conf
```

Pay particular attention to the section on encrypted passwords. If you have enabled clear-text passwords in the guest operating system, be sure that `smb.conf` is set up to use clear-text passwords. Similarly, if you are using encrypted passwords, you must have the same setting in the guest operating system and in `smb.conf`.

Note: Using Samba printer sharing with virtual machines is not supported. Consult the man pages for guidance on configuring Samba for printing.

Sample smb.conf for Host-Only Networking

The following sample Samba configuration file is for use with host-only networking. This configuration is for the 2.0.6 version of Samba installed by GSX Server. The configuration files are placed in `/etc/vmware/vmnet1/smb` by default.

```
# This is the VMware(TM) Samba configuration file. You should read the
# smb.conf(5) manual page in order to understand the options listed
# here. Samba has a huge number of configurable options
# most of which are not shown in this example
#
# Any line that starts with a ; (semicolon) or a # (hash)
# is a comment and is ignored. In this example we will use a #
# for commentary and a ; for parts of the config file that you
# may wish to enable
#
#
# Configuration file for Samba 2.0.6 vmware-[sn]mbd operating on
# vmnet1.
#
# This file was generated by the VMware configuration
# program and modified for this document.
#
# If you modify it, it will be backed up the next time you run the
# configuration program.

# Global settings
[global]

# This should be polled at install time from the private subnet created by
# vmware-config.pl
socket address = 192.168.183.1
interfaces = vmnet1
bind interfaces only = yes

workgroup = WORKGROUP
netbios name = HOSTNAME
server string = VMware host-only

security = user
encrypt passwords = yes

# Note: Printers not loaded in this example. Resource definitions commented
# below.
; load printers = yes

socket options = TCP_NODELAY SO_RCVBUF=8192 SO_SNDBUF=8192
```

```
# VMware extension to use a different shared memory access key on each
# Samba server running on this host
sysv shm key = /dev/vmnet1

; log file = /etc/vmware/vmnet1/smb/var/log.smb
; log level = 1
; max log size in KB
; max log size = 50

lock directory = /etc/vmware/vmnet1/smb/var/locks

smb passwd file = /etc/vmware/vmnet1/smb/private/smbpasswd

codepage dir = /usr/lib/vmware/smb/codepages

dns proxy = no

# Shared resources

# Home directories
[homes]
comment = Home directories
browseable = no
writable = yes

# Printers
;[printers]
; comment = All printers
; path = /var/lpd
; browseable = no
; guest ok = no
; writable = no
; printable = yes

;[HostFS]
; comment = VMware host filesystem
; path = /
; public = no
; writeable = yes
; printable = no
```

Sample smb.conf for Bridged Networking

The following sample Samba configuration file is for use with bridged networking. This configuration file is based on the 2.0.7 version of Samba and assumes that you are using your existing Samba server, as provided with your host computer's Linux distribution. The configuration file is placed in `/etc` by default.

```
# This is the main Samba configuration file. You should read the
# smb.conf(5) manual page in order to understand the options listed
# here. Samba has a huge number of configurable options
# most of which are not shown in this example
#
# Any line that starts with a ; (semicolon) or a # (hash)
# is a comment and is ignored. In this example we will use a #
# for commentary and a ; for parts of the config file that you
```

```
# may wish to enable
#
# NOTE: Whenever you modify this file you should run the command
# "testparm" to check that you have not many any basic syntactic
# errors.

# Global Settings

[global]

interfaces = eth0

workgroup = WORKGROUP
netbios name = HOSTNAME
server string = Samba Host Box

# Note: Printers not loaded in this example. Resource definitions commented
# below.
; printcap name = lpstat
; load printers = yes
; printing = cups

socket options = TCP_NODELAY SO_RCVBUF=8192 SO_SNDBUF=8192

log file = /var/log/samba/log.%m
max log size = 50

security = user
encrypt passwords = yes
smb passwd file = /etc/smbpasswd

dns proxy = no

preserve case = yes
short preserve case = yes
default case = lower
; case sensitive = no

# Shared Resources

[homes]
comment = Home Directories
browseable = yes
writable = yes

;[printers]
; comment = All Printers
; path = /var/spool/samba
; browseable = yes
; guest ok = yes
; writable = no
; printable = yes
; create mode = 0700
; print command = lpr-cups -P %p -o raw %s -r # using client side
; printer drivers.
; print command = lpr-cups -P %p %s # using cups own drivers (use
; generic PostScript on clients).
; lpq command = lpstat -o %p
```

```
; lprm command = cancel %p-%j

;[system]
; comment = System share
; path = /
; valid users = username
; public = no
; browsable = yes
; writable = yes
; printable = no
```

Adding User Names and Passwords to the GSX Server Samba Password File

You must be sure the Samba password file includes entries for all users of the virtual machine who will access the host's file system. The user names and passwords in the Samba password file must be the same as those used for logging on to the guest operating system.

You may add user names and passwords to the GSX Server Samba password file at any time from a terminal window on your Linux host computer.

1. Log on to the root account.
   `su -`
2. Run the GSX Server Samba password command.
   `vmware-smbpasswd vmnet1 -a <username>`
   `<username>` is the user name you want to add. Follow the instructions on the screen.

   Note: `vmware-smbpasswd` is based on the standard Samba password program. If you are familiar with the options used in `smbpasswd`, you may use any of them in `vmware-smbpasswd`.
3. Log off of the root account.
   `exit`

You may receive an error message that says
`Unknown virtual interface "vmnet1"`
This indicates your machine is not using the GSX Server Samba server.

If your installation of GSX Server does not include the GSX Server Samba server and you want to set it up, log on to the root account on your host computer (`su -`), then run `vmware-config.pl` from a terminal on the host. The configuration program asks
`Do you want this script to automatically configure your system to allow your virtual machines to access the host file system?`
Answer `yes`.

Caution: In order to configure GSX Server correctly, the `vmware-config.pl` configuration program requires all virtual machines to be shut down. The program shuts down any running virtual machines automatically.

If You Are Already Running Samba

If you already have Samba running on your Linux host, you should not install the GSX Server Samba server when you are installing GSX Server on your host.

The configuration program prompts you

```
Do you want this script to automatically configure your
system to allow your virtual machines to access the host
file system?
```

Answer `no`.

Be sure to modify your Samba configuration so it includes the IP subnet used by the GSX Server virtual Ethernet adapter, VMnet1.

To determine what subnet is being used by VMnet1, run

```
/sbin/ifconfig vmnet1
```

You must be sure the Samba password file includes entries for all users of the virtual machine who will access the host's file system. The user names and passwords in the Samba password file must be the same as those used for logging on to the guest operating system.

You may add user names and passwords to the Samba password file at any time from a terminal window on your Linux host computer.

1. Log on to the root account.
   ```
   su -
   ```
2. Run the Samba password command.
   ```
   smbpasswd -a <username>
   ```
   `<username>` is the user name you want to add. Follow the instructions on the screen.
3. Log off of the root account.
   ```
   exit
   ```

Using a Samba Server for Both Bridged and Host-Only Networks

You may use the Samba server of your choice — either the existing Samba server from your host operating system's distribution or the one provided with GSX Server — for both host-only and bridged networking. To do so, you must modify one parameter

in the `smb.conf` file. You can define the `interface` parameter so your Samba server serves multiple interfaces. An example of this is:

```
interface = eth0 vmnet1
```

This example tells the Samba server that it is to listen to and use both the `eth0` and `vmnet1` interfaces — the interfaces used by bridged and host-only networking, respectively.

Using GSX Server's Samba with an Existing Installation

It may also be possible to run both your existing Samba server and the GSX Server Samba server at the same time. In order to do this, your current Samba server must be version 2.0.6 or higher and must be configured correctly. However, this approach is not recommended.

To determine the version of your Samba server, run

```
smbd -V
```

If you want to try running both Samba servers at the same time, use this sample `smb.conf` file as a basis for configuring the regular Samba server on your host computer.

Sample smb.conf for Running Two Samba Servers at the Same Time

```
; This file is the recommended smb.conf file for your
; normal Samba server if you want to run it concurrently
; (which we don't advise) with the VMware Samba server.
;
; Your normal samba server should be at least v 2.0.6
;
; Note that you will need to insert specific information
; for your system at several points indicated in the file
; by <text in angle brackets>.
;
; --------------
;
; Larmor samba server configuration
;
; Global settings
[global]
;
; Identity
;
; Allow several Samba servers on the same machine
interfaces = <your real subnet>/<your real netmask>
bind interfaces only = yes
; Workgroup the host belongs to
workgroup = VMware
; SMB name of the host (the hostname by default)
netbios name = <your Windows name>
; Description of the host
server string = Linux running Samba 2.0.6
;
```

```
; Access
;
; Allow connections from
; hosts allow = <your real subnet>/<your real netmask>
; Authentication scheme
security = user
encrypt passwords = yes
;
; Options
;
; Automatically load the printer list (from /etc/printcap
; by default)
load printers = yes
; Gives better performance
socket options = TCP_NODELAY SO_RCVBUF=8192 SO_SNDBUF=8192
;
; Files and directories
;
; Max log size in KB
max log size = 1024
; Locks
lock directory = /var/samba
; SMB passwords
smb passwd file = /etc/samba/smbpasswd
;
; Name browsing
;
; Allow the host to participate in master browser
; elections
local master = yes
; Force a local browser election upon startup
; We need that otherwise it takes a long time before the
; windows network is browsable
preferred master = yes
; Do not try to resolve SMB names via DNS
dns proxy = no

; Shared resources
;
; Home directories
[homes]
comment = Home directories
browseable = no
writable = yes
; Printers
;[printers]
; comment = All printers
; path = /var/lpd
; browseable = no
; guest ok = no
; writable = no
; printable = yes
[Slash]
comment = Whole filesystem
path = /
public = no
writeable = yes
printable = no
```

CHAPTER 8

Configuring Devices

The following sections describe how to use various devices with a virtual machine:

- Using Parallel Ports on page 290
- Using Serial Ports on page 296
- Keyboard Mapping on a Linux Host on page 310
- Using USB Devices in a Virtual Machine on page 319
- Connecting to a Generic SCSI Device on page 325

Using Parallel Ports

VMware GSX Server supports a partial emulation of bidirectional PS/2-style ports.

On Linux hosts, GSX Server requires that the parallel port "PC-style hardware" option (`CONFIG_PARPORT_PC`) be built and loaded as a kernel module (that is, it must be set to `"m"`). GSX Server is unable to use parallel port devices if `CONFIG_PARPORT_PC` is built directly (compiled) into the kernel. This limitation exists because `CONFIG_PARPORT_PC` does not correctly export its symbols.

The following sections describe how to use parallel ports with GSX Server:

- About Parallel Ports on page 290
- Adding a Parallel Port in a Virtual Machine on page 290
- Configuring a Parallel Port on a Linux Host on page 292
- Special Notes for the Iomega Zip Drive on page 295

About Parallel Ports

Parallel ports are used by a variety of devices, including printers, scanners, dongles and disk drives.

Currently, GSX Server provides only partial emulation of PS/2 hardware. Specifically, interrupts requested by a device connected to the physical port are not passed to the virtual machine. Also, the guest operating system cannot use DMA (direct memory access) to move data to or from the port. For this reason, not all devices that attach to the parallel port are guaranteed to work correctly.

You can attach up to three parallel ports to a virtual machine. The virtual parallel port can connect to a parallel port or a file on the host operating system.

Adding a Parallel Port in a Virtual Machine

If the virtual machine is configured with a parallel port, most guest operating systems automatically detect it at installation time and install the required drivers. Some operating systems, including Linux, Windows NT and Windows 2000, automatically detect the ports at boot time. Others, like Windows 95 and Windows 98, do not.

To add a parallel port to the virtual machine's configuration, complete the following steps with the virtual machine powered off. You can add the device from the console or from the management interface.

Note: In a Windows 95 or Windows 98 guest, run the guest operating system's Add New Hardware Wizard (**Start** > **Settings** > **Control Panel** > **Add New Hardware**) after you add the port and let Windows detect the new device.

Adding a Parallel Port from the Console

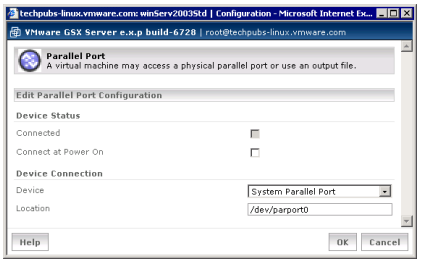
1. Open the virtual machine settings editor. Choose **VM** > **Settings**.
2. Click **Add** to start the New Hardware Wizard.
3. Select **Parallel Port**, then click **Next**.
4. Make the appropriate selection to use a physical parallel port or connect the virtual parallel port to a file, then click **Next**.
5. If you selected **Use physical parallel port on the host**, choose the port from the **Physical parallel port** list.

   If you selected **Output file**, enter the path and filename in the **Output file** field, or browse to the location of the file.

   Under **Device status**, the default setting is **Connect at power on**. Clear the check box if you do not want the parallel port device to be connected when the virtual machine powers on.
6. Click **Finish** to install the virtual parallel port, then click **OK** to save the configuration and close the virtual machine settings editor.

Adding a Parallel Port from the Management Interface

1. On the Hardware page, click **Add Device**. The Add Device Wizard starts.
2. Click **Parallel Port**. The Parallel Port page appears.



3. To connect this virtual machine to the host's parallel port when the virtual machine is powered on, check **Connect at Power On**.
4. Specify whether to connect to the host's physical parallel port or to an output file. In the **Device** list, select **System Parallel Port** or **Output File**.
5. Enter the location of the device in the **Location** field. For example, the host's parallel port could be `LPT1` or `/dev/parport0`.

   Note: If you are connecting with a Windows console to add a physical parallel port to a virtual machine on a remote Linux host, be sure to specify a Linux

device name here, such as `/dev/parport0`. If you are connecting with a Linux console to add a physical parallel port to a virtual machine on a remote Windows host, be sure to specify a Windows device name here, such as `LPT1`.

6. Click **OK** to add the parallel port.

Configuring a Parallel Port on a Linux Host

For the parallel port to work properly in a guest, it must first be configured properly on the host. Most issues involving parallel port functionality are a result of the host configuration. Check these areas of concern: the version of your Linux kernel, your device access permissions and the required modules.

Parallel Ports and Linux 2.6.x Kernels

Be sure that PC Style Hardware (CONFIG_PARPORT_PC) is loaded as a module as mentioned at the beginning of Using Parallel Ports on page 290. If you are using a 2.6.x kernel, the modules that provide parallel port functionality are `parport_pc` and `ppdev`.

To see if these modules are installed and running on your system, run the `lsmod` command as the root user. You can also look at the `/proc/modules` file for the same list.

With a 2.6.x kernel, loading `parport_pc` does not load both modules. If neither of the listed parallel port modules is running, use this command:

```
modprobe parport_pc && modprobe ppdev
```

This command inserts both modules needed for a parallel port.

If you continue to see problems, it is possible that the `lp` module is running. If it is, the virtual machine cannot use the parallel port correctly. To remove the `lp` module, run this command as the root user:

```
rmmod lp
```

You should also ensure that the line referring to the `lp` module in the `/etc/modules.conf` or `/etc/conf.modules` file is removed or commented out by inserting a hash character (#) at the beginning of the line. The name of the configuration file depends on the Linux distribution you are using. When you reboot the host after removing this line, the configuration file no longer starts the `lp` module.

To ensure that the proper modules for the parallel port are loaded at boot time, add this line to the `/etc/modules.conf` or `/etc/conf.modules` file:

```
alias parport_lowlevel parport_pc
```

Linux kernels in the 2.6.x series also use a special arbitrator that allows access to the parallel port hardware. If the parallel port is in use by the host, the guest cannot use it. If a virtual machine is using the parallel port, the host and any users accessing the host are not given access to the device. GSX Server puts a lock on the device, and this lock restricts access so only the virtual machine can use the port.

You can choose **VM** > **Removable Devices** to disconnect the parallel port from the virtual machine and reconnect it.

Parallel Ports and Linux 2.4.x Kernels

Be sure that PC Style Hardware (`CONFIG_PARPORT_PC`) is loaded as a module, as mentioned at the beginning of Using Parallel Ports on page 290. If you are using a 2.4.x kernel, the modules that provide parallel port functionality are `parport`, `parport_pc` and `ppdev`.

To see if these modules are installed and running on your system, run the `lsmod` command as the root user. These three modules should be included in the listing of running modules. You can also look at the `/proc/modules` file for the same list.

To load the proper modules, run this command:

```
insmod -k <modulename>
```

If none of the listed parallel port modules is running, use this command:

```
insmod -k parport_pc
```

This command inserts the three modules needed for a parallel port.

If you continue to see problems, it is possible that the `lp` module is running. If it is, the virtual machine cannot use the parallel port correctly. To remove the `lp` module, run this command as the root user:

```
rmmod lp
```

You should also ensure that the line referring to the `lp` module in the `/etc/modules.conf` or `/etc/conf.modules` file is removed or commented out by inserting a hash character (#) at the beginning of the line. The name of the configuration file depends on the Linux distribution you are using. When you reboot the host after removing this line, the configuration file no longer starts the `lp` module.

To ensure that the proper modules for the parallel port are loaded at boot time, add this line to the `/etc/modules.conf` or `/etc/conf.modules` file:

```
alias parport_lowlevel parport_pc
```

Linux kernels in the 2.4.x series also use a special arbitrator that allows access to the parallel port hardware. If the parallel port is in use by the host, the guest cannot use it.

If a virtual machine is using the parallel port, the host and any users accessing the host are not given access to the device. GSX Server puts a lock on the device, and this lock restricts access so only the virtual machine can use the port.

You can choose **VM** > **Removable Devices** to disconnect the parallel port from the virtual machine and reconnect it.

Parallel Ports and Linux 2.2.x Kernels

The 2.2.x kernels that support parallel ports use the `parport`, `parport_pc` and `vmppuser` modules. Be sure that PC Style Hardware (`CONFIG_PARPORT_PC`) is loaded as a module, as mentioned at the beginning of Using Parallel Ports on page 290. The `vmppuser` module is supplied by GSX Server to give virtual machines user-level access to the parallel port.

To see if these modules are installed and running on your system, run the `lsmod` command as the root user. These three modules should be included in the listing of running modules. You can also look at the `/proc/modules` file for the same list.

To load the proper modules, run this command:

```
insmod -k <modulename>
```

If none of the listed parallel port modules is running, use this command:

```
insmod -k parport_pc
```

This command inserts the three modules needed for a parallel port.

If you continue to see problems, it is possible that the `lp` module is running. If it is, the virtual machine cannot use the parallel port correctly. To remove the `lp` module, run this command as the root user:

```
rmmod lp
```

You should also ensure that the line referring to the `lp` module in the `/etc/modules.conf` or `/etc/conf.modules` file is removed or commented out by inserting a hash character (`#`) at the beginning of the line. The name of the configuration file depends on your Linux distribution. When you reboot the host after removing this line, the configuration file no longer starts the `lp` module.

To ensure that the proper modules for the parallel port are loaded at boot time, add this line to the `/etc/modules.conf` or `/etc/conf.modules` file:

```
alias parport_lowlevel parport_pc
```

Device Permissions

Some Linux distributions by default do not grant the virtual machine access to the `lp` and `parport` devices. In most of these cases, the owner of the device is `root` and

the associated group is `lp`. To allow the GSX Server user to access the device, add the user to the associated group. To view the owner and group of the device, run this command:

```
ls –la /dev/parport0
```

The third and fourth columns of the output show the owner and group, respectively.

To add the user to the device group, edit the `/etc/group` file. On the line starting with `lp`, which defines the `lp` group, add the GSX Server user's user name. You must make this change as the root user. The following line provides an example for a user whose user name is `userj`.

```
lp::7:daemon,lp,userj
```

The next time the user logs on to the host, the changes take effect.

Special Notes for the Iomega Zip Drive

On Windows 95 or Windows 98, use of older drivers for the Iomega Zip drive may cause the guest operating system to lock up intermittently at boot time or during installation of the guest operating system. The newest Iomega drivers work reliably in our tests. They are available at *www.iomega.com/software/index.html*.

Using Serial Ports

The following sections describe how to use serial ports with GSX Server:

- Using a Serial Port on the Host Computer on page 296
- Using a File on the Host Computer on page 298
- Connecting an Application on the Host to a Virtual Machine on page 300
- Connecting Two Virtual Machines on page 303
- Special Configuration Options for Advanced Users on page 306
- Examples: Debugging over a Virtual Serial Port on page 307

A GSX Server virtual machine can use up to four virtual serial ports. The virtual serial ports can be configured in several ways.

- You can connect a virtual serial port to a physical serial port on the host computer.
- You can connect a virtual serial port to a file on the host computer.
- You can make a direct connection between two virtual machines or between a virtual machine and an application running on the host computer.

You can also select whether to connect the virtual serial port when you power on the virtual machine.

Using a Serial Port on the Host Computer

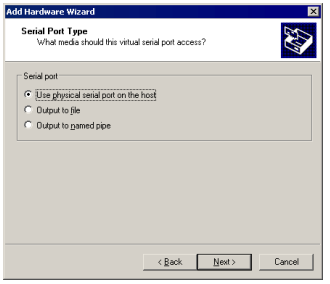
You can set up the virtual serial port in a virtual machine to use a physical serial port on the host computer. This is useful, for example, if you want to use an external modem or a hand-held device in your virtual machine.

To install a virtual serial port that connects to a physical serial port on the host computer, take the following steps with the virtual machine powered off. You can add the device from the console or from the management interface.

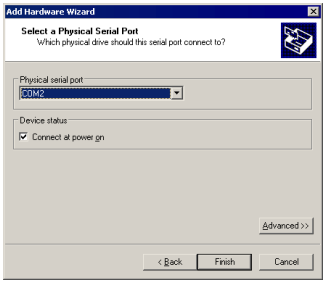
Adding a Serial Port from the Console

1. Open the virtual machine settings editor (choose **VM** > **Settings**).
2. Click **Add** to start the Add Hardware Wizard.

3. Select **Serial Port**, then click **Next**. The Serial Port Type screen appears.



4. Select **Use physical serial port on the host**, then click **Next**. The Select a Physical Serial Port screen appears.



5. Choose the port on the host computer that you want to use for this serial connection. By default, the device status setting is **Connect at power on**. You may deselect this setting if you wish.

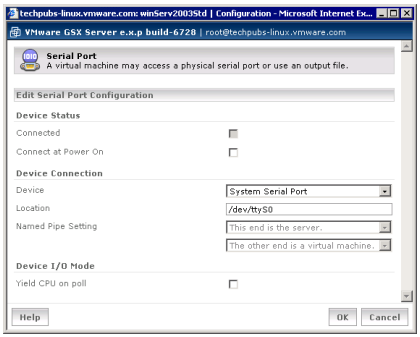
   Note: If you are connecting with a Windows console to add a physical serial port to a virtual machine on a remote Linux host, be sure to specify a Linux device name here, such as `/dev/ttyS0`. If you are connecting with a Linux console to add a physical serial port to a virtual machine on a remote Windows host, be sure to specify a Windows device name here, such as COM1.

   Click **Advanced** if you want to configure this serial port to use polled mode. This option is of interest primarily to developers who are using debugging tools that communicate over a serial connection. For more information, see Special Configuration Options for Advanced Users on page 306.

6. Click **Finish**, then click **OK** to close the virtual machine settings editor.

Adding a Serial Port from the Management Interface

1. On the Hardware page, click **Add Device**. The Add Device Wizard starts.
2. Click **Serial Port**. The Serial Port page appears.



3. To connect this virtual machine to the host's serial port when the virtual machine is powered on, check **Connect at Power On**.
4. Connect to the host's physical serial port. In the **Device** list, select **System Serial Port**.
5. Enter the location of the device in the **Location** field. For example, the host's serial port could be `COM1` or `/dev/ttyS0`.

   Note: If you are connecting with a Windows console to add a physical serial port to a virtual machine on a remote Linux host, be sure to specify a Linux device name here, such as `/dev/ttyS0`. If you are connecting with a Linux console to add a physical serial port to a virtual machine on a remote Windows host, be sure to specify a Windows device name here, such as COM1.
6. Check **Yield CPU on Poll** if you want to configure this serial port to use polled mode. This option is of interest primarily to developers who are using debugging tools that communicate over a serial connection. For more information, see Special Configuration Options for Advanced Users on page 306.
7. Click **OK** to add the serial port.

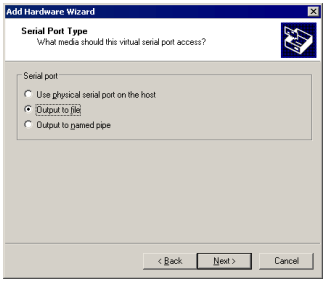
Using a File on the Host Computer

You can set up the virtual serial port in a virtual machine to send its output to a file on the host computer. This is useful, for example, if you want to capture the data that a program running in the virtual machine sends to the virtual serial port, or if you need a quick way to transfer a file from the guest to the host.

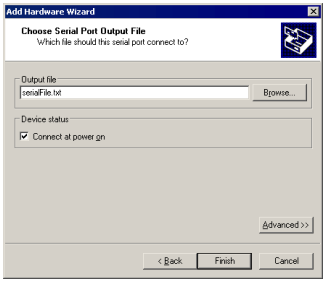
To install a virtual serial port that connects to a file on the host computer, take the following steps with the virtual machine powered off. You can add the device from the console or from the management interface.

Connecting to an Output File from the Console

1. Open the virtual machine settings editor (**VM** > **Settings**).
2. Click **Add** to start the Add Hardware Wizard.
3. Select **Serial Port**, then click **Next**. The Serial Port Type screen appears.



4. Select **Output to file**, then click **Next**. The Choose Serial Port Output File screen appears.



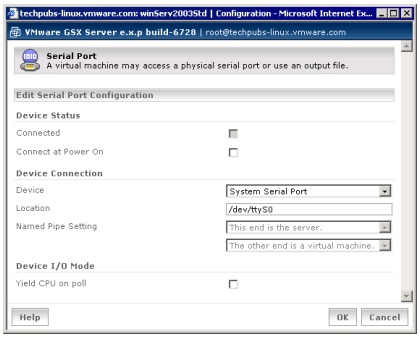
5. Browse to the file on the host computer that you want to use to store the output of the virtual serial port. By default, the device status setting is **Connect at power on**. You may deselect this setting if you wish.

   Click **Advanced** if you want to configure this serial port to use polled mode. This option is of interest primarily to developers who are using debugging tools that communicate over a serial connection. For more information, see Special Configuration Options for Advanced Users on page 306.

6. Click **Finish**, then click **OK** to close the virtual machine settings editor.

Connecting to an Output File from the Management Interface

1. On the Hardware page, click **Add Device**. The Add Device Wizard starts.
2. Click **Serial Port**. The Serial Port page appears.



3. To connect this virtual machine to the device when the virtual machine is powered on, check **Connect at Power On**.
4. Connect to a file on the host. In the **Device** list, select **Output File**.
5. Enter the location of the file in the **Location** field.
6. Check **Yield CPU on Poll** if you want to configure this device to use polled mode. This option is of interest primarily to developers who are using debugging tools that communicate over a serial connection. For more information, see Special Configuration Options for Advanced Users on page 306.
7. Click **OK** to add the device.

Connecting an Application on the Host to a Virtual Machine

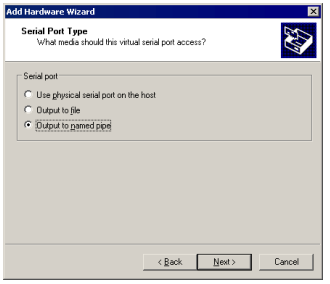
You can set up the virtual serial port in a virtual machine to connect to an application on the host computer. This is useful, for example, if you want to use an application on the host to capture debugging information sent from the virtual machine's serial port.

To install a direct serial connection between an application on the host and a virtual machine, take the following steps with the virtual machine powered off. You can add the device from the console or from the management interface.

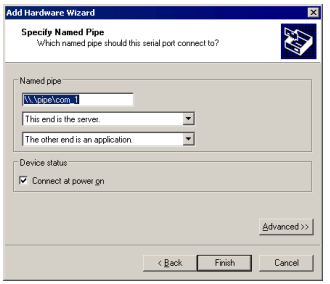
Connecting to an Application from the Console

1. Open the virtual machine settings editor (**VM** > **Settings**).
2. Click **Add** to start the Add Hardware Wizard.

3. Select **Serial Port**, then click **Next**. The Serial Port Type screen appears.



4. Select **Output to named pipe**, then click **Next**. The Specify Named Pipe screen appears.



5. Use the default pipe name, or enter another pipe name of your choice.

   For a serial pipe on a Windows host, the pipe name must follow the form `\\.\pipe\<namedpipe>` — that is, it must begin with `\\.\pipe\`.

   For a serial pipe on a Linux host, enter `/tmp/<socket>` or another Unix socket name of your choice.

   Note: If you are using a Windows console to connect to a virtual machine on a remote Linux host, be sure to specify a Linux pipe name here, such as `/tmp/<pipe>`. If you are using a Linux console to connect to a virtual machine on a remote Windows host, be sure to specify a Windows pipe name here, such as `\\.\pipe\<namedpipe>`.

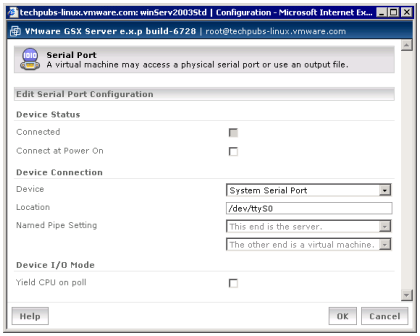
6. Select **This end is the server** or **This end is the client**. In general, select **This end is the server** if you plan to start this end of the connection first.
7. Select **The other end is an application**.
8. The default device status setting is **Connect at power on**. You may deselect this setting if you wish.

Click **Advanced** if you want to configure this serial port to use polled mode. This option is of interest primarily to developers who are using debugging tools that communicate over a serial connection. For more information, see Special Configuration Options for Advanced Users on page 306.

9. Click **Finish**, then click **OK** to save your configuration and close the virtual machine settings editor.
10. On your host computer, configure the application that communicates with the virtual machine to use the same pipe or Unix socket name.

Connecting to an Application from the Management Interface

1. On the Hardware page, click **Add Device**. The Add Device Wizard starts.
2. Click **Serial Port**. The Serial Port page appears.



3. To connect this virtual machine to the device when the virtual machine is powered on, check **Connect at Power On**.
4. Connect to a named pipe on the host. In the **Device** list, select **Named Pipe**.
5. Enter the location of the file in the **Location** field.

   For a serial pipe on a Windows host, the pipe name must follow the form `\\.\pipe\<namedpipe>` — that is, it must begin with `\\.\pipe\`.

   For a serial pipe on a Linux host, enter `/tmp/<socket>` or another Unix socket name of your choice.

   Note: If you are using a Windows console to connect to a virtual machine on a remote Linux host, be sure to specify a Linux pipe name here, such as `/tmp/<pipe>`. If you are using a Linux console to connect to a virtual machine on a remote Windows host, be sure to specify a Windows pipe name here, such as `\\.\pipe\<namedpipe>`.

6. Select **This end is the server** or **This end is the client**. In general, select **This end is the server** if you plan to start this end of the connection first.
7. Select **The other end is an application**.
8. Check **Yield CPU on Poll** if you want to configure this device to use polled mode. This option is of interest primarily to developers who are using debugging tools that communicate over a serial connection. For more information, see Special Configuration Options for Advanced Users on page 306.
9. Click **OK** to add the device.

Connecting Two Virtual Machines

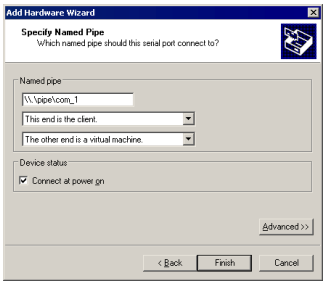
You can set up the virtual serial ports in two virtual machines to connect to each other. This is useful, for example, if you want to use an application in one virtual machine to capture debugging information sent from the other virtual machine's serial port.

To install a direct serial connection between two virtual machines (a server and a client), take the following steps with the virtual machine powered off. You can add the device from the console or from the management interface.

Note: Make sure you performs these steps **twice**: once for the **server** virtual machine and once for the **client** virtual machine.

Connecting Two Virtual Machines from the Console

1. Connect to the **server** virtual machine with a console.
2. Open the virtual machine settings editor (**VM** > **Settings**).
3. Click **Add** to start the Add Hardware Wizard.
4. Select **Serial Port**, then click **Next**.
5. Select **Output to named pipe**, then click **Next**. The Specify Named Pipe screen appears.



6. Use the default pipe name, or enter another pipe name of your choice.

   For a serial pipe on a GSX Server for Windows host, the pipe name must follow the form `\\.\pipe\<namedpipe>` — that is, it must begin with `\\.\pipe\`.

   For a serial pipe on a GSX Server for Linux host, enter `/tmp/<socket>` or another Unix socket name of your choice.

   Note: If you are using a Windows console to connect to a virtual machine on a remote Linux host, be sure to specify a Linux pipe name here, such as `/tmp/<pipe>`. If you are using a Linux console to connect to a virtual machine on a remote Windows host, be sure to specify a Windows pipe name here, such as `\\.\pipe\<namedpipe>`.

7. For the **server** virtual machine, select **This end is the server**.

   For the **client** virtual machine, select **This end is the client**.

8. Select **The other end is a virtual machine**.

9. By default, the device status setting is **Connect at power on**. You may deselect this setting if you wish.

   Click **Advanced** if you want to configure this serial port to use polled mode. This option is of interest primarily to developers who are using debugging tools that communicate over a serial connection. For more information, see Special Configuration Options for Advanced Users on page 306.

10. Click **Finish**, then click **OK** to save your configuration and close the virtual machine settings editor.

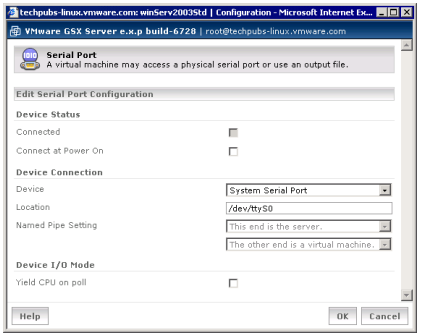
11. Repeat these steps for the **client** virtual machine.

Connecting Two Virtual Machines from the Management Interface

1. Connect to the **server** virtual machine with the management interface.

2. On the Hardware page, click **Add Device**. The Add Device Wizard starts.

3. Click **Serial Port**. The Serial Port page appears.



4. To connect this virtual machine to the device when the virtual machine is powered on, check **Connect at Power On**.
5. Connect to a named pipe on the host. In the **Device** list, select **Named Pipe**.
6. Enter the location of the file in the **Location** field.

   For a serial pipe on a Windows host, the pipe name must follow the form `\\.\pipe\<namedpipe>` — that is, it must begin with `\\.\pipe\`.

   For a serial pipe on a Linux host, enter `/tmp/<socket>` or another Unix socket name of your choice.

   Note: If you are using a Windows console to connect to a virtual machine on a remote Linux host, be sure to specify a Linux pipe name here, such as `/tmp/<pipe>`. If you are using a Linux console to connect to a virtual machine on a remote Windows host, be sure to specify a Windows pipe name here, such as `\\.\pipe\<namedpipe>`.
7. For the **server** virtual machine, select **This end is the server**.

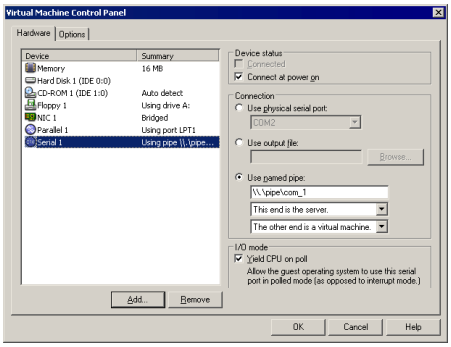
   For the **client** virtual machine, select **This end is the client**.
8. Select **The other end is a virtual machine**.
9. Check **Yield CPU on Poll** if you want to configure this device to use polled mode. This option is of interest primarily to developers who are using debugging tools that communicate over a serial connection. For more information, see Special Configuration Options for Advanced Users on page 306.
10. Click **OK** to add the parallel port.
11. Repeat these steps for the **client** virtual machine.

Special Configuration Options for Advanced Users

Two special configuration options are available for serial connections between a virtual machine and the host or between two virtual machines. These options are of interest primarily to developers who are using debugging tools that communicate over a serial connection.

Improving Processor Performance When Debugging

The first option must be set in the virtual machine settings editor (**VM** > **Settings** > **Serial Port**) or the VMware Management Interface (**Configure Hardware** > **Edit** next to the serial port). This option is useful when the serial port is being used by the guest operating system in polled mode as opposed to interrupt mode. Polled mode causes the virtual machine to consume a disproportionate share of processor time. This makes the host and other guests run sluggishly.



To restore performance for applications on the host, check the **Yield CPU on poll** check box. This configuration option forces the affected virtual machine to yield processor time if the only task it is trying to do is poll the virtual serial port.

Changing the Input Speed of the Serial Connection

To use the second option, power off the virtual machine and close the console window, then use a text editor to add the following line to your virtual machine's configuration file (`.vmx`):

```
serial<n>.pipe.charTimePercent = <x>
```

This option is useful if you want to squeeze every possible bit of speed from your serial connection over a pipe to the virtual machine. In principle, there is no limit on the output speed — the speed at which the virtual machine sends data through the virtual serial port. In practice, the output speed depends on how fast the application at the other end of the pipe reads data inbound to it.

<n> is the number of the serial port, starting from 0. So the first serial port is serial0.

<x> is any positive integer. It specifies the time taken to transmit a character, expressed as a percentage of the default speed set for the serial port in the guest operating system. For example, a setting of 200 forces the port to take twice as long per character, or send data at half the default speed. A setting of 50 forces the port to take only half as long per character, or send data at twice the default speed.

You should first use the guest operating system to configure the serial port for the highest setting supported by the application you are running in the virtual machine.

Once the serial port speed is set appropriately in the guest operating system, experiment with this setting. Start with a value of 100 and gradually decrease it until you find the highest speed at which your connection works reliably.

Examples: Debugging over a Virtual Serial Port

You can use Debugging Tools for Windows (WinDbg) or Kernel Debugger (KD) to debug kernel code in a virtual machine over a virtual serial port. You can download Debugging Tools for Windows from the Windows DDK Web site at *www.microsoft.com/whdc/devtools/debugging/default.mspx*.

The following two examples illustrate how to use a virtual serial port to debug kernel code in a virtual machine:

- With the debugging application on the GSX Server host (Windows hosts only)
- With the debugging application in another virtual machine on the same GSX Server host (useful on a Linux host and can also be done on a Windows host)

Either of these methods lets you debug kernel code on one system, without requiring two physical computers, a modem or serial cable.

Debugging an Application in a Virtual Machine from the Windows Host

In this example, you have kernel code to debug in a virtual machine (called the target virtual machine) and are running WinDbg or KD on your Windows host.

To prepare the target virtual machine, follow the steps in Connecting an Application on the Host to a Virtual Machine on page 300. Make sure you configure the virtual machine's virtual serial port as follows:

- Select **This end is the server**
- Click **Advanced**, then under **I/O Mode**, select the **Yield CPU on poll** check box, as the kernel in the target virtual machine uses the virtual serial port in polled mode, not interrupt mode

To prepare the host, make sure you have a recent version of Debugging Tools for Windows — one that supports debugging over a pipe. You need version 4.0.18.0 or higher.

When you are ready to begin, complete the following steps:

1. Power on the virtual machine.
2. Check to make sure the serial port is connected. Choose **VM** > **Removable Devices**. On that menu, **serial<n>** should be reported as **\\.\pipe\<namedpipe>** (on Windows hosts) or **/tmp/<socket>** (on Linux hosts). If the serial port is not connected, choose the virtual serial port, then **Connect**.
3. On the host, open a Command Prompt window and do one of the following:
   - If you are using `WinDbg`, type the following:

     ```
     windbg -k com:port=\\.\pipe\<namedpipe>,pipe
     ```

   - If you are using `KD`, type the following:

     ```
     kd -k com:port=\\.\pipe\<namedpipe>,pipe
     ```

   Then press Enter to start debugging.

Debugging an Application in a Virtual Machine from another Virtual Machine

In this situation, you have kernel code to debug in a virtual machine (called the target virtual machine) and are running Debugging Tools for Windows (`WinDbg`) or Kernel Debugger (`KD`) in another virtual machine (called the debugger virtual machine) on the same host.

This setup is useful if you are running GSX Server on a Linux host. The debugger virtual machine must be running Debugging Tools for Windows (`WinDbg`) or Kernel Debugger (`KD`) in a Windows guest operating system.

To prepare the target virtual machine, follow the steps for the **server** virtual machine in Connecting Two Virtual Machines on page 303. Make sure when you configure the target virtual machine's virtual serial port that you select the **Yield CPU on poll** check box, as the kernel in the target virtual machine uses the virtual serial port in polled mode, not interrupt mode.

To prepare the debugger virtual machine, make sure you have downloaded Debugging Tools for Windows. Then follow the steps for the **client** virtual machine in Connecting Two Virtual Machines on page 303.

When you are ready to continue, complete the following steps:

1. Power on both virtual machines.

2. Check to make sure the serial port is connected. Choose **VM** > **Removable Devices**. If the serial port is not connected, choose the virtual serial port, then **Connect**.
3. In the debugger virtual machine, start debugging with `WinDbg` or `KD` normally.

Keyboard Mapping on a Linux Host

This section addresses the following issues and provides additional details on keyboard mapping in Linux:

- My (language-specific) keyboard is not supported by GSX Server.
- Some of the keys on my keyboard don't work right in the virtual machine.
- My keyboard works fine when I run a virtual machine locally, but not when I run the same virtual machine with a remote X server.

The following sections describe keyboard mapping on a Linux host:

- Quick Answers on page 310
- The Longer Story on page 310
- V-Scan Code Table on page 314

Quick Answers

If your keyboard works correctly with a local X server, and you just want the same behavior with a remote X server (which is also an XFree86 server running on a PC), just power off the virtual machine and close the console, then add the line

```
xkeymap.usekeycodeMapIfXFree86 = true
```

to the virtual machine configuration file or to `~/.vmware/config`. Make this change on the host machine, where you run the virtual machine, not on the machine with the remote X server.

If you are using an XFree86-based server that GSX Server does not recognize as an XFree86 server, use this instead:

```
xkeymap.usekeycodeMap = true
```

If you are using an XFree86 server running locally, and the keyboard does not work correctly, please report the problem by submitting a support request at *www.vmware.com/requestsupport*.

The Longer Story

Unfortunately, keyboard support for the PC (virtual or otherwise) is a complex affair. To explain it properly, we have to start with some background information — greatly simplified.

Pressing a key on the PC keyboard generates a scan code based roughly on the position of the key. For example, the Z key on a German keyboard generates the same code as the Y key on an English keyboard, because they are in the same position on

the keyboard. Most keys have one-byte scan codes, but some keys have two-byte scan codes with prefix `0xe0`.

Internally, GSX Server uses a simplified version of the PC scan code that is a single nine-bit numeric value, called a v-scan code. A v-scan code is written as a three-digit hexadecimal number. The first digit is 0 or 1. For example, the left-hand Ctrl key has a one-byte scan code (`0x1d`); its v-scan code is `0x01d`. The right-hand Ctrl key scan code is two bytes (`0xe0`, `0x1d`); its v-scan code is `0x11d`.

An X server uses a two-level encoding of keys. An X key code is a one-byte value. The assignment of key codes to keys depends on the X server implementation and the physical keyboard. As a result, an X application normally cannot use key codes directly. Instead, the key codes are mapped into keysyms that have names like space, escape, x and 2. The mapping can be controlled by an X application via the function `XChangeKeyboardMapping()` or by the program `xmodmap`. To explore keyboard mappings, you can use `xev`, which shows the key codes and keysyms for keys typed into its window.

To recap, a key code corresponds roughly to a physical key, while a keysym corresponds to the symbol on the key top. For example, with an XFree86 server running on a PC, the Z key on the German keyboard has the same key code as the Y key on an English keyboard. The German Z keysym, however, is the same as the English Z keysym, and different from the English Y keysym.

For an XFree86 server on a PC, there is a one-to-one mapping from X key codes to PC scan codes (or v-scan codes, which is what GSX Server really uses). GSX Server takes advantage of this fact. When it is using an XFree86 server on the local host, it uses the built-in mapping from X key codes to v-scan codes. This mapping is keyboard independent and should be correct for most, if not all, languages. In other cases (not an XFree86 server or not a local server), GSX Server must map keysyms to v-scan codes, using a set of keyboard-specific tables.

Key code mapping is simple, automatic and foolproof. (Keysym mapping is more complex and described later.) However, because the program cannot tell whether a remote server is running on a PC or on some other kind of computer, it errs on the safe side and uses key code mapping only with local X servers. This is often too conservative and has undesirable effects. Luckily, this and other behavior related to key code-mapping can be controlled by powering off the virtual machine and closing the console, then using a text editor to add configuration settings to the virtual machine's configuration file.

- `xkeymap.usekeycodeMapIfXFree86 = true`
  Use key code mapping if you are using an XFree86 server, even if it is remote.

- `xkeymap.usekeycodeMap = true`
  Always use key code mapping regardless of server type.
- `xkeymap.nokeycodeMap = true`
  Never use key code mapping.
- `xkeymap.keycode.<code> = <v-scan code>`
  If using key code mapping, map key code `<code>` to `<v-scan code>`. In this example, `<code>` must be a decimal number and `<v-scan code>` should be a C-syntax hexadecimal number (for example, `0x001`).

The easiest way to find the X key code for a key is to run `xev` or `xmodmap -pk`. Most of the v-scan codes are covered in the V-Scan Code Table on page 314. The keysym mapping tables described in this section are also helpful.

Use this feature to make small modifications to the mapping. For example, to swap left Ctrl and Caps Lock, use the following lines:

```
xkeymap.keycode.64 = 0x01d # X Caps_Lock -> VM left ctrl
xkeymap.keycode.37 = 0x03a # X Control_L -> VM caps lock
```

These configuration lines can be added to the individual virtual machine configuration, to your personal GSX Server configuration (`~/.vmware/config`), or even to the host-wide (`/etc/vmware/config`) or installation-wide (usually `/usr/local/lib/vmware/config`) configuration.

When key code mapping cannot be used (or is disabled), GSX Server maps keysyms to v-scan codes. It does this using one of the tables in the `xkeymap` directory in the GSX Server installation (usually `/usr/local/lib/vmware`).

Which table you should use depends on the keyboard layout. The normal distribution includes tables for PC keyboards for the United States and a number of European countries and languages. And for most of these, there are both the 101-key (or 102-key) and the 104-key (or 105-key) variants.

GSX Server automatically determines which table to use by examining the current X keymap. However, its decision-making process may sometimes fail. In addition, each mapping is fixed and may not be completely right for any given keyboard and X key code-to-keysym mapping. For example, a user may have swapped Ctrl and Caps Lock using `xmodmap`. This means the keys are swapped in the virtual machine when using a remote server (keysym mapping) but unswapped when using a local server (key code mapping).

Therefore, keysym mapping is necessarily imperfect. To make up for this defect, you can change most of the behavior using configuration settings:

- `xkeymap.language = <keyboard-type>`
  Use this if GSX Server has a table in `xkeymap` for your keyboard but can't detect it. `<keyboard-type>` must be one of the tables in the `xkeymap` directory. (See above for location.) However, the failure to detect the keyboard probably means the table isn't completely correct for you.
- `xkeymap.keysym.<sym> = <v-scan code>`
  If you use keysym mapping, map keysym `<sym>` to `<v-scan code>`. When you do, `<sym>` must be an X keysym name and `<v-scan code>` should be a C-syntax hexadecimal number (for example, `0x001`).

  The easiest way to find the keysym name for a key is to run `xev` or `xmodmap -pk`.

  The X header file `/usr/X11R6/include/X11/keysymdef.h` has a complete list of keysyms. (The name of a keysym is the same as its C constant without the `XK_` prefix.) Most v-scan codes are in the V-Scan Code Table on page 314.

  The `xkeymap` tables themselves are also helpful. Use them to fix small errors in an existing mapping.
- `xkeymap.fileName = <file-path>`
  Use the keysym mapping table in `<file-path>`. A table is a sequence of configuration lines of the form
  `<sym> = <v-scan code>`
  where `<sym>` is an X keysym name, and `<v-scan code>` is a C-syntax hexadecimal number (for example, `0x001`). (See the explanation of `xkeymap.keysym` above for tips on finding the keysyms and v-scan codes for your keyboard.)

  Compiling a complete keysym mapping is difficult. It is best to start with an existing table and make small changes.

V-Scan Code Table

These are the v-scan codes for the 104-key U.S. keyboard:

Symbol	Shifted symbol	Location	V-scan code
Esc			0x001
1	!		0x002
2	@		0x003
3	#		0x004
4	$		0x005
5	%		0x006
6	^		0x007
7	&		0x008
8	*		0x009
9	(		0x00a
0	)		0x00b
-	_		0x00c
=	+		0x00d
Backspace			0x00e
Tab			0x00f
Q			0x010
W			0x011
E			0x012
R			0x013
T			0x014
Y			0x015
U			0x016
I			0x017
O			0x018
P			0x019
[	{		0x01a
]	}		0x01b
Enter			0x01c
Ctrl		left	0x01d

Symbol	Shifted symbol	Location	V-scan code
A			0x01e
S			0x01f
D			0x020
F			0x021
G			0x022
H			0x023
J			0x024
K			0x025
L			0x026
;			0x027
'			0x028
`			0x029
Shift		left	0x02a
\	|		0x02b
Z			0x02c
X			0x02d
C			0x02e
V			0x02f
B			0x030
N			0x031
M			0x032
,	<		0x033
.	>		0x034
/	?		0x035
Shift		right	0x036
*		numeric pad	0x037
Alt		left	0x038
Space bar			0x039
Caps Lock			0x03a
F1			0x03b

Symbol	Shifted symbol	Location	V-scan code
F2			0x03c
F3			0x03d
F4			0x03e
F5			0x03f
F6			0x040
F7			0x041
F8			0x042
F9			0x043
F10			0x044
Num Lock		numeric pad	0x045
Scroll Lock			0x046
Home	7	numeric pad	0x047
Up arrow	8	numeric pad	0x048
PgUp	9	numeric pad	0x049
-		numeric pad	0x04a
Left arrow	4	numeric pad	0x04b
5		numeric pad	0x04c
Right arrow	6	numeric pad	0x04d
+		numeric pad	0x04e
End	1	numeric pad	0x04f
Down arrow	2	numeric pad	0x050
PgDn	3	numeric pad	0x051

Symbol	Shifted symbol	Location	V-scan code
Ins	0	numeric pad	0x052
Del		numeric pad	0x053
F11			0x057
F12			0x058
Break	Pause		0x100
Enter		numeric pad	0x11c
Ctrl		right	0x11d
/		numeric pad	0x135
SysRq	Print Scrn		0x137
Alt		right	0x138
Home		function pad	0x147
Up arrow		function pad	0x148
Page Up		function pad	0x149
Left arrow		function pad	0x14b
Right arrow		function pad	0x14d
End		function pad	0x14f
Down arrow		function pad	0x150
Page Down		function pad	0x151
Insert		function pad	0x152
Delete		function pad	0x153
Windows		left	0x15b

Symbol	Shifted symbol	Location	V-scan code
Windows		right	0x15c
Menu			0x15d

The 84-key keyboard has a Sys Req key on the numeric pad:

Symbol	Shifted symbol	Location	V-scan code
Sys Req		numeric pad	0x054

Keyboards outside the U.S. usually have an extra key (often < > or < > |) next to the left shift key:

Symbol	Shifted symbol	Location	V-scan code
<	>		0x056

Using USB Devices in a Virtual Machine

The following sections describe how to use USB devices in a virtual machine:

- Notes on USB Support on page 319
- Adding a USB Controller on page 320
- Connecting USB Devices on page 321
- Using USB with a Windows Host on page 321
- Replacing USB 2.0 Drivers on a Windows 2000 Host on page 322
- Using USB with a Linux Host on page 322
- Who Has Control over a USB Device? on page 322
- Disconnecting USB Devices from a Virtual Machine on page 324
- Human Interface Devices on page 324

VMware GSX Server provides a two-port USB 1.1 controller. You can use up to two USB devices in your virtual machine if both your host operating system and your guest operating system support USB. If your host computer supports USB 2.0 devices, you can use those devices in the virtual machine.

Note: Linux kernels older than 2.2.17 do not support USB.

Although your host operating system must support USB, you do not need to install device-specific drivers for your USB devices in the host operating system if you want to use those devices only in the virtual machine.

On a Windows 2000 host computer with USB 2.0 support, be sure you are using the Microsoft USB 2.0 driver for the USB controller. Third-party USB 2.0 drivers, such as those provided by some motherboard manufacturers, are not supported. For notes on replacing the third-party drivers, see Replacing USB 2.0 Drivers on a Windows 2000 Host on page 322.

Notes on USB Support

We have tested a variety of USB devices with this release. In general, if the guest operating system has appropriate drivers, you should be able to use PDAs, printers, storage (disk) devices, scanners, MP3 players, PC radios, digital cameras and memory card readers.

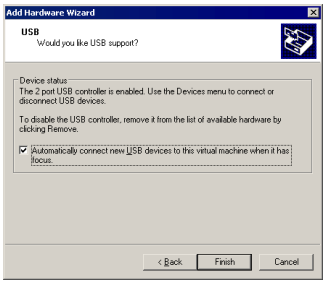
Modems and certain streaming data devices, such as speakers and Web cams, do not work properly.

Adding a USB Controller

The USB controller is disabled by default in all virtual machines created with GSX Server. If you want to add a USB controller to the virtual machine's configuration, complete the following steps with the virtual machine powered off. You can add the device from the console or from the management interface.

Adding a USB Controller from the Console

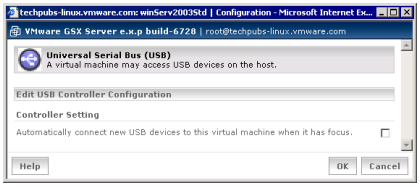
1. Open the virtual machine settings editor. Choose **VM** > **Settings**.
2. Click **Add** to start the New Hardware Wizard. Click **Next**. The Hardware Type screen appears.
3. Select **USB Controller**, then click **Next**. The USB screen appears.


Add Hardware Wizard
USB
Would you like USB support?
Device status
The 2 port USB controller is enabled. Use the Devices menu to connect or disconnect USB devices.
To disable the USB controller, remove it from the list of available hardware by clicking Remove.
Automatically connect new USB devices to this virtual machine when it has focus.
< Back
Finish
Cancel

4. If you don't want new USB devices to be automatically connected to the virtual machine, clear the check box.
5. Click **Finish** to install the virtual USB controller, then click **OK** to save the configuration and close the virtual machine settings editor.

Adding a USB Controller from the Management Interface

1. On the Hardware page, click **Add Device**. The Add Device Wizard starts.
2. Click **USB Controller**. The Universal Serial Bus (USB) page appears.


techpubs-linux.vmware.com: winServ2003Std | Configuration - Microsoft Internet Ex...
VMware GSX Server e.x.p build-6728 | root@techpubs-linux.vmware.com
Universal Serial Bus (USB)
A virtual machine may access USB devices on the host.
Edit USB Controller Configuration
Controller Setting
Automatically connect new USB devices to this virtual machine when it has focus.
Help
OK
Cancel

3. To connect USB devices to this virtual machine when the virtual machine is selected in the console, check the check box.
4. Click **OK** to add the USB controller.

Connecting USB Devices

When a virtual machine is running, its window is the active window and a USB device is plugged into the host computer, the device automatically connects to the guest instead of the host. This autoconnect feature can be disabled in the USB Controller panel of the virtual machine settings editor (**VM** > **Settings**). If all of the virtual machine's USB ports are already occupied when it is trying to connect automatically to a new device, a dialog box gives you a choice: you can either disconnect one of the existing USB devices to free its port, or ignore the new device, allowing the device to connect to the host.

Choose **VM** > **Removable Devices** to connect specific USB devices to your virtual machine. You can connect up to two USB devices at a time. If the physical USB devices are connected to the host computer through a hub, the virtual machine sees only the USB devices, not the hub.

There is a menu item for each of the USB ports. Move the mouse over one of these items to see a cascading menu of devices that are plugged into your host computer and available for use. To connect a device to the virtual machine, click its name.

If a device is already connected to that port, click the name of a new device to release the first device and connect the new one.

To release a connected device, click **None** on the cascading menu for the port to which it is connected.

If you physically plug a new device into the host computer and the autoconnect feature does not connect it to a virtual machine, the device is initially connected to the host. Its name is also added to the **Removable Devices** submenu so you can connect it to the virtual machine manually.

Using USB with a Windows Host

When a particular USB device is connected to a virtual machine for the first time, the host detects it as a new device named VMware USB Device and installs the appropriate VMware driver.

On a Windows Server 2003 host, User confirmation is required in the Found New Hardware Wizard. Select the default action — **Install the software automatically**. Once the software is installed, the guest operating system detects the USB device and searches for a suitable driver.

When you are synchronizing a PDA, such as a Palm handheld or Handspring Visor, to a virtual machine for the first time, the total time required to load the VMware USB device driver in the host and the PDA driver in the guest may exceed the device's connection timeout value. This causes the device to disconnect itself from the

computer before the guest can synchronize with it. If this occurs, let the guest finish installing the PDA driver, dismiss any connection error warnings, then try synchronizing the PDA again. The second attempt should succeed.

Replacing USB 2.0 Drivers on a Windows 2000 Host

To use GSX Server on a Windows 2000 host that has USB 2.0 ports, you must use the Microsoft USB 2.0 drivers for the USB controller in the host operating system. If your host operating system is using a third-party driver — a driver supplied by your motherboard vendor, for example — you must replace it.

Take the following steps to check the provider of your driver:

1. Go to the Device Manager. Right-click **My Computer**, choose **Properties**, click the **Hardware** tab, then click **Device Manager**.
2. Expand the listing for Universal Serial Bus controllers.
3. Right-click the listing for the controller and choose **Properties**.
4. Click the **Driver** tab. If the driver provider shown on that page is Microsoft, you have the correct driver already.

If the driver provider is not Microsoft, download the latest USB driver for your host operating system from the Microsoft Web site and follow the Microsoft instructions to install it. Details are available in Microsoft knowledge base article 319973.

Using USB with a Linux Host

On Linux hosts, GSX Server uses the USB device file system to connect to USB devices. In most Linux systems that support USB, the USB device file system is at `/proc/bus/usb`. If your host operating system uses a different path to the USB device file system, you can change it in the virtual machine settings editor (**VM** > **Settings** > **USB**). Enter the correct path in the **Path to usbdevfs** field.

Who Has Control over a USB Device?

Only one computer — host or guest — can have control of a USB device at any one time.

Device Control on a Windows Host

When you connect a device to a virtual machine, it is "unplugged" from the host or from the virtual machine that previously had control of the device. When you disconnect a device from a virtual machine, it is "plugged in" to the host.

Caution: You need to take a special step to disconnect USB network and storage devices from the host. There is a system tray icon called Eject Hardware on Windows

2000 and Safely Remove Hardware on Windows Server 2003. Use this icon to disconnect the device from the host before connecting it to a virtual machine.

Note: When you connect a USB network or storage device in a virtual machine, you may see a message on your host that says the device can be removed safely. This is normal behavior, and you can simply dismiss the dialog box. However, do **not** remove the device from your physical computer. GSX Server automatically transfers control of the device to the virtual machine.

Under some circumstances, if a USB storage device is in use on the host (for example, one or more files stored on the device are open on the host), an error appears in the virtual machine when you try to connect to the device. You must let the host complete its operation or close any application connected to the device on the host, then connect to the device in the virtual machine again.

Device Control on a Linux Host

On Linux hosts, guest operating systems can use devices that are not already in use by the host — that is, devices that are not claimed by a host operating system driver.

If your device is in use by the host and you try to connect it to the guest using the VM > Removable Devices menu, a dialog box appears, informing you that there is a problem connecting to the device.

To disconnect the device from the host, you must unload the device driver. You can unload the driver manually as root (`su -`) using the `rmmod` command. Or, if the driver was automatically loaded by `hotplug`, you can disable it in the `hotplug` configuration files in the `/etc/hotplug` directory. See your Linux distribution's documentation for details on editing these configuration files.

A related issue sometimes affects devices that rely on automatic connection (as PDAs often do).

If you have successfully used autoconnection to connect the device to your virtual machine, then experience problems with the connection to the device, take the following steps:

1. Disconnect and reconnect the device. You can either unplug it physically, then plug it back in or use the VM > Removable Devices menu to disconnect it and reconnect it.
2. If you see a dialog box warning that the device is in use, disable it in the `hotplug` configuration files in the `/etc/hotplug` directory.

Disconnecting USB Devices from a Virtual Machine

Before unplugging a USB device or using the **Removable Devices** submenu to disconnect it from a virtual machine, be sure it is in a safe state.

You should follow the procedures the device manufacturer specifies for unplugging the device from a physical computer. This is true whether you are physically unplugging it, moving it from host to virtual machine, moving it between virtual machines or moving it from virtual machine to host.

This is particularly important with data storage devices (a Zip drive, for example). If you move a data storage device too soon after saving a file and the operating system has not actually written the data to the disk, you can lose data.

Human Interface Devices

USB human interface devices, such as the keyboard and mouse, are not handled through the virtual machine's USB controller. Instead, they appear in the virtual machine as a standard PS/2 keyboard and mouse, even though they are plugged into USB ports on the host.

Connecting to a Generic SCSI Device

The following sections describe how to use generic SCSI devices in a virtual machine:

- Device Support in Guest Operating Systems on page 325
- Adding a Generic SCSI Device to a Virtual Machine on page 326
- Generic SCSI on a Windows Host Operating System on page 328
- Generic SCSI on a Linux Host Operating System on page 330

Generic SCSI lets a virtual machine run any SCSI device that is supported by the guest operating system in the virtual machine. Generic SCSI gives the guest operating system direct access to SCSI devices connected to the host, such as scanners, tape drives and tape changers.

Device Support in Guest Operating Systems

In theory, generic SCSI is completely device independent, but VMware has discovered it is sensitive to the guest operating system, device class and specific SCSI hardware. We encourage you to try any SCSI hardware you want to use and report problems to VMware technical support.

Preparing a Windows XP Guest Operating System to Use SCSI Devices

To use SCSI devices in a Windows XP virtual machine, you need a special SCSI driver available from the download section of the VMware Web site at *www.vmware.com/download*. Follow the instructions on the Web site to install the driver.

Preparing a Windows NT 4.0 Guest Operating System to Use SCSI Devices

Generic SCSI devices use the virtual Mylex® (BusLogic) BT/KT-958 compatible host bus adapter provided by the virtual machine. Some guest operating systems guide you through installing the drivers after you install the first SCSI device in the virtual machine. On Windows NT 4.0, however, you may need to install the driver manually, if it is not already installed for a virtual SCSI disk. You should do so before you add a generic SCSI device.

To install the BusLogic driver in a Windows NT 4.0 guest, have your Windows NT installation CD available and follow these steps.

1. Open the SCSI Adapters control panel.

   Start > **Settings** > **Control Panel** > **SCSI Adapters**
2. Click the **Drivers** tab.
3. Click **Add**.

4. In the list of vendors on the left, select **BusLogic**.
5. In the list of drivers on the right, select **BusLogic MultiMaster PCI SCSI Host Adapters**.
6. Click **OK**.
7. Insert the Windows NT CD when you are prompted. Click **OK**.
8. Reboot the guest operating system when you are prompted.

Preparing a Windows Me, Windows 98 or Windows 95 Guest Operating System to Use SCSI Devices

If you are using generic SCSI devices in a Windows 95, Windows 98 or Windows Me guest operating system and are experiencing problems with the devices, download the latest Mylex (BusLogic) BT/KT-958 compatible host bus adapter from *www.lsilogic.com*. This driver overrides what Windows chooses as the best driver, but it corrects known problems.

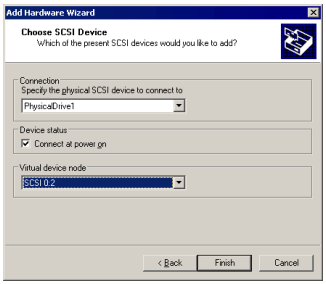
Adding a Generic SCSI Device to a Virtual Machine

You can add generic SCSI devices to your virtual machine in the virtual machine settings editor. The virtual machine settings editor lets you map virtual SCSI devices to physical generic SCSI devices on the host.

To add a new generic SCSI device to a virtual machine, make sure the virtual machine is powered off, then complete the appropriate steps below.

Adding a Generic SCSI Device to a Virtual Machine from the Console

1. Launch a VMware Virtual Machine Console and select the virtual machine.
2. Choose **VM** > **Settings**. The virtual machine settings editor opens.
3. Click **Add** to start the Add Hardware Wizard. Select **Generic SCSI Device**, then click **Next**.



4. Choose the name of the physical device you want to use.

Then choose the virtual device node where you want this device to appear in the virtual machine.

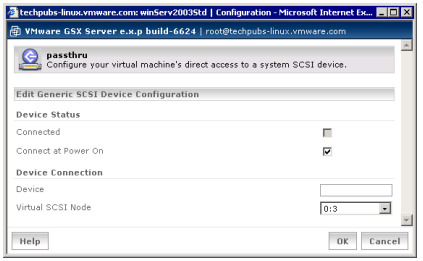
A check box under **Device status** allows you to specify whether or not the device should be connected each time the virtual machine is powered on.

Note: On a Windows host, the device should appear in the form CdRom0, Scanner0, Tape0 or Changer0. If you do not see a listing for the device, you may need to add the device to the virtual machine manually. See Generic SCSI on a Windows Host Operating System on page 328.

5. Click **Finish** to install the new device.
6. Click **OK** to save the configuration and close the virtual machine settings editor.

Adding a Generic SCSI Device to a Virtual Machine from the Management Interface

1. On the Hardware page, click **Add Device**. The Add Device Wizard starts.
2. Click **Generic SCSI Device**. The SCSI Device page appears.



3. To connect this virtual machine to the server's SCSI device when the virtual machine is powered on, check **Connect at Power On**.
4. In the **Device** field, specify the name of the device, such as `Scanner0` (for a device on a Windows host) or `/dev/sga` (for a device on a Linux host).
5. Specify the virtual device node. Select the appropriate SCSI ID in the **Virtual SCSI Node** list.
6. Click **OK** to add the device.

Generic SCSI on a Windows Host Operating System

Using the SCSI Generic driver in Windows, GSX Server allows your guest operating system to operate generic SCSI devices — including scanners, tape drives and other data storage devices — in a virtual machine.

Adding a Generic SCSI Device Not Detected by GSX Server (Advanced Users)

When adding a generic SCSI device to a virtual machine, if GSX Server does not display the device you want to add (for example, scanners on a Windows 2000 host or some tape backup devices), you need to add the device manually to the virtual machine's configuration file (`.vmx`).

Reasons GSX Server cannot detect a device include:

- A driver for that device is not installed on the host.
- A driver on the host prevents the device from being detected.
- The virtual machine uses a device for which there are no drivers available to the host operating system.

Before you attempt the steps below, first verify that the device driver is installed on the host. If the driver is not installed, install it then see if the device appears correctly to GSX Server. If it does not appear correctly, or if you cannot or do not want to install the driver on the host, then you need to add the device manually to the virtual machine.

When adding a device manually to the virtual machine, use `scsiX:Y` notation to refer to the device on the host instead of a device name GSX Server uses like `CdRom0`. For this type of notation, `X` is the SCSI bus on which the device is located on the host and `Y` is the target ID the device uses on the host.

Caution: Adding a device in this manner is recommended for advanced users only.

Caution: Before you add the device, you must disable the original SCSI device driver on the host. Some Windows operating systems do not process the send command from the adapter if the device driver is owning the device.

There are a few circumstances requiring you to add the device manually. Follow the steps that match your circumstance. In each case, power off the virtual machine, then open the virtual machine's configuration file (`.vmx`) in a text editor and make the changes as described below.

1. The virtual machine does not contain any SCSI adapters or devices, or you want to add a generic SCSI device to a new virtual SCSI adapter in the virtual machine.

   In this case, to add the device to the virtual machine, you need to add the following lines to the virtual machine's configuration file:

```
scsiZ:Y.present = "true"
scsiZ:Y.deviceType = "scsi-passthru"
scsiZ:Y.fileName = "scsiX:Y"
```

Define `X`, `Y` and `Z` as follows:

- `X` is the SCSI bus the device uses on the host system.
- `Y` is the target ID the device uses in the virtual machine **and** on the host. Use the same target ID in the virtual machine that the host already uses for the device to allow the device to work correctly.
- `Z` is the SCSI bus the device uses in the virtual machine.

2. The virtual machine has a SCSI adapter and a SCSI device and you want to use the same device as a generic SCSI device.

   In this case, to configure the device as a generic SCSI device, you need to add the following lines to the virtual machine's configuration file:

   ```
   scsiZ:Y.deviceType = "scsi-passthru"
   scsiZ:Y.fileName = "scsiX:Y"
   ```

   Define `X`, `Y` and `Z` as follows:

   - `X` is the SCSI bus the device uses on the host system.
   - `Y` is the target ID the device uses in the virtual machine **and** on the host. Use the same target ID in the virtual machine that the host already uses for the device to allow the device to work correctly.
   - `Z` is the SCSI bus the device uses in the virtual machine.

3. The virtual machine has a SCSI adapter and generic SCSI device, but GSX Server does not recognize the device in the Add Hardware Wizard.

   In this case, you need to look for a line in the configuration file that looks like:

   ```
   scsiZ:Y.fileName = "<deviceName>"
   ```

   Change the line to:

   ```
   scsiZ:Y.fileName = "scsiX:Y"
   ```

   Define `X`, `Y` and `Z` as follows:

   - `X` is the SCSI bus the device uses on the host system.
   - `Y` is the target ID the device uses in the virtual machine **and** on the host. Use the same target ID in the virtual machine that the host already uses for the device to allow the device to work correctly.
   - `Z` is the SCSI bus the device uses in the virtual machine.

For example, if the problematic device is a CD-ROM drive, the entry in the configuration file might be:

```
scsi0:4.fileName = "CdRom0"
```

If the device on the host is located on bus 2 with target ID 4, you should change this line to:

```
scsi0:4.fileName = "scsi2:4"
```

The target ID the device uses in the virtual machine must be the same as the target ID the device uses on the host system.

4. You added a generic SCSI device to the virtual machine's configuration file (`.vmx`) as instructed in step 3, but GSX Server does not recognize the device in the Add Hardware Wizard. You may experience this issue with tape drives and tape changers.

   In this case, look for the line in the configuration file that looks like:

   ```
   scsiZ:Y.fileName = "scsiX:Y"
   ```

   Change the line to:

   ```
   scsiZ:Y.fileName = "<deviceName>"
   ```

   Example device names include `Tape0` and `Changer0`.

   Another alternative you can try is to uninstall or disable the device driver on the host and use the `"scsiX:Y"` notation in the configuration file.

Note: The SCSI bus is assigned a number by the host operating system after all IDE buses have been assigned numbers. For example, if you have 2 IDE buses, they are numbered 0 and 1. The first SCSI bus is assigned bus number 2. In the example above, you use 2 for `X`.

If you cannot determine the SCSI bus number on your own, you can try using a third-party tool like `winobj` (which you can download for free from *www.sysinternals.com*) to determine this information.

The device target ID is usually set by some jumpers or switches on the device. Refer to the owner's manual for the device for information on how to determine the target ID.

Generic SCSI on a Linux Host Operating System

Using the SCSI Generic driver in Linux, GSX Server allows your guest operating system to operate generic SCSI devices within a virtual machine. The SCSI Generic driver sets up a mapping for each SCSI device in `/dev`. Each entry starts with `sg` (for the SCSI Generic driver) followed by a letter. For example, `/dev/sga` is the first generic SCSI device.

Each entry corresponds to a SCSI device, in the order specified in `/proc/scsi/scsi`, from the lowest device ID on the lowest adapter to the highest device ID on the lowest adapter, and so on to the highest device ID on the highest adapter. Do not enter `/dev/st0` or `/dev/scd0`.

Note: When setting up a generic SCSI device in the virtual machine settings editor, as described later in this section, you specify the device you wish to install in the virtual machine by typing its `/dev/sg` entry in the **Connection** field.

Requirements

Generic SCSI requires version 2.1.36 of the SCSI Generic (`sg.o`) driver, which comes with kernel 2.2.14 and higher.

Avoiding Concurrent Access to a Generic SCSI Device

Under Linux some devices — specifically tape drives, disk drives and CD-ROM drives — already have a designated `/dev` entry (traditionally, `st`, `sd` and `scd`, respectively). When the SCSI Generic driver is installed, Linux also identifies these devices with corresponding `sg` entries in `/dev` — in addition to their traditional entries. GSX Server ensures that multiple programs are not using the same `/dev/sg` entry at the same time but cannot always ensure that multiple programs are not using the `/dev/sg` and the traditional `/dev` entry at the same time. It is important that you do not attempt to use the same device in both host and guest. This can cause unexpected behavior and may cause loss or corruption of data.

Permissions on a Generic SCSI Device

You must have read and write permissions on a given generic SCSI device in order to use the device within a virtual machine, even if the device is a read-only device such as a CD-ROM drive. These devices typically default to root-only permissions. Your administrator should create a group with access to read and write to these devices, then add the appropriate users to that group.

CHAPTER 9

Video and Sound

The following sections provide information on configuring the video display and sound for VMware GSX Server.

- Setting Screen Color Depth in a Virtual Machine on page 334
- Using Full Screen Mode on a Linux Host on page 336
- Configuring Sound on page 337

Setting Screen Color Depth in a Virtual Machine

The number of screen colors available in the guest operating system depends on the screen color setting of the host operating system.

Virtual machines support

- 16-color (VGA) mode
- 8-bit pseudocolor
- 16 bits per pixel (16 significant bits per pixel)
- 32 bits per pixel (24 significant bits per pixel)

If the host is in 15-bit color mode, the guest operating system's color setting controls offer 15-bit mode in place of 16-bit mode.

If the host is in 24-bit color mode, the guest operating system's color setting controls offer 24-bit mode in place of 32-bit mode.

If you run a guest operating system set for a greater number of colors than your host operating system is using, you can encounter various problems. In some cases, the colors in the guest are not correct. In others, the guest operating system is not able to use a graphical interface.

To fix these problems, you can either increase the number of colors available on the host or decrease the number of colors used in the guest.

For best performance, use the same number of colors in the guest and on the host.

The following sections describe changing the color depth on the host and in a virtual machine:

- Changing Screen Color Depth on the Host on page 334
- Changing Screen Color Depth in the Virtual Machine on page 335

Changing Screen Color Depth on the Host

If you choose to change the color settings on your host operating system, you should first shut down all guest operating systems, power off the virtual machines and close the console.

Follow standard procedures for changing the color settings on your host operating system, then restart the console and the virtual machines.

Changing Screen Color Depth in the Virtual Machine

If you choose to change the color settings in the guest operating system, the approach you use depends on the combination of host and guest you are using.

Follow the normal process for changing screen colors in your guest operating system. In a Windows guest, the Display Properties control panel offers only those settings that are supported.

In a Linux or FreeBSD guest, you must change the color depth before you start the X server, or restart the X server after you make the changes.

Using Full Screen Mode on a Linux Host

When you switch a virtual machine into full screen mode, GSX Server changes the full screen display resolution to better match the resolution set in the guest operating system. On a Linux host, GSX Server uses the VidMode or DGA2 extension from the XFree86 Project or Xfs from Xig to match the host resolution to the one requested by the guest running in the virtual machine.

In a few cases, GSX Server may not find the best resolution.

When GSX Server switches into full screen mode, it can choose only those resolutions that are already configured on your host.

If a virtual machine runs at a resolution that does not match a mode listed in the X server configuration, then for full screen mode GSX Server chooses the closest larger mode (and uses black borders) or else simply does not offer full screen mode at all.

It is possible to have bad modes configured in the `XF86Config` file on your host. If your host's X server configuration was automatically generated, or if you never tested all modes with your current monitor and video card, it is possible that some enabled modes do not work with your monitor. However, the mode-switching code in GSX Server has no way of knowing this and a virtual machine that tries to use a resolution with a bad mode line can cause your monitor to fail to display correctly.

If this happens, immediately leave full screen mode by pressing Ctrl-Alt, then fix your X server configuration and restart the X server. However, if the only problem is that the image is off center or is not quite the right size on the monitor, you can usually correct it using the controls on your monitor. Note that most modern monitors are capable of storing separate settings for each resolution, so changing the settings for a new mode should not impair the settings for the host resolution.

Configuring Sound

GSX Server provides a sound device compatible with the Sound Blaster AudioPCI adapter and supports sound in Windows 95, Windows 98, Windows Me, Windows NT, Windows 2000, Windows XP, Windows Server 2003 and Linux guest operating systems. The GSX Server sound device is disabled by default and must be installed using the virtual machine settings editor (**VM** > **Settings**).

Sound support includes PCM (pulse code modulation) output and input. For example, you can play `.wav` files, MP3 audio and Real Media audio. MIDI output from Windows guests is supported through the Windows software synthesizer. MIDI input is not supported, and no MIDI support is available for Linux guests.

Windows 2000, Windows XP and most recent Linux distributions automatically detect the sound device and install appropriate drivers for it.

The following sections describe installing sound drivers in some Windows guest operating systems.

Installing Sound Drivers in a Windows Server 2003 Guest Operating System

Windows Server 2003 does not ship with the drivers for the Sound Blaster AudioPCI adapter. You can install the drivers from a Windows 2000 installation CD-ROM. For information on installing these drivers, see the VMware knowledge base article at *www.vmware.com/support/kb/enduser/std_adp.php?p_faqid=1115*.

Installing Sound Drivers in Windows 9x and Windows NT Guest Operating Systems

Windows 95, Windows 98, Windows 98SE and Windows NT 4.0 do not have drivers for the Sound Blaster AudioPCI adapter. To use sound in these guest operating systems, you must download the driver from the Creative Labs Web site (*www.creative.com*) and install it in the guest operating system.

Creative Labs has a number of Web sites serving various regions of the world. The adapter name varies, depending on the region, but usually includes PCI 128.

CHAPTER 10

Performance Tuning for Virtual Machines

The following sections offer suggestions for getting the best performance from VMware GSX Server and your virtual machines:

- Allocating Memory to a Virtual Machine on page 340
- Improving Performance for Guest Operating Systems on page 343

Allocating Memory to a Virtual Machine

The following sections describe how you can allocate memory to a virtual machine:

- Configuring Virtual Machine Memory from a Console on page 341
- Configuring Virtual Machine Memory from the Management Interface on page 342

GSX Server allows you to allocate a portion of the GSX Server host memory to each virtual machine. By adjusting this setting, you can affect the virtual machine's performance.

You set the size of an individual virtual machine's memory in the virtual machine settings editor or the VMware Management Interface. The minimum size of the memory for the virtual machine should be set based on the recommendations of the operating system provider.

When you create a new virtual machine, the wizard sets what VMware believes are reasonable defaults for the memory size of a virtual machine, based on the type of the guest operating system and the amount of memory in the host computer.

The actual size that should be given to a virtual machine depends on a few practical considerations:

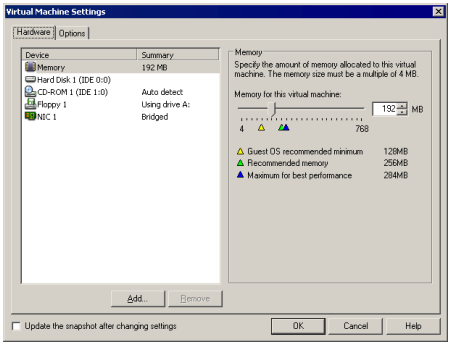
- What kinds of applications will run in the virtual machine.
- What other virtual machines will contend with this virtual machine for memory resources.
- What applications will run on the host at the same time as the virtual machine.
- The total amount of host memory that all running virtual machines can use; for more information, see Specifying How Much RAM is Used by All Running Virtual Machines in the *VMware GSX Server Administration Guide*.
- The file system where the virtual machine is stored. You cannot allocate more than 2000MB of memory to a virtual machine if it is stored on a file system that cannot support files larger than 2GB, such as FAT16. You will not be able to power on such a virtual machine. Further, you cannot allocate more than 2000MB of memory to a virtual machine if it is stored on a FAT32 file system, even though it does support files up to 4GB in size.

For more information on host memory use, see Understanding Memory Usage in the *VMware GSX Server Administration Guide*.

Configuring Virtual Machine Memory from a Console

To set the size of an individual virtual machine's memory from the VMware Virtual Machine Console, complete the following steps.

1. Connect to the virtual machine with a console.
2. Open the virtual machine settings editor (**VM** > **Settings**). The virtual machine settings editor opens with the Memory tab selected.



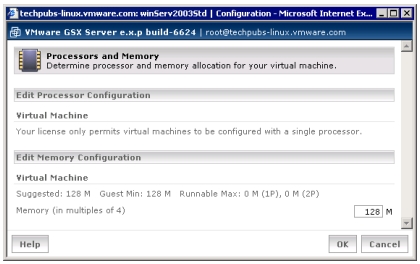
3. Allocate memory to the virtual machine. Use the slider or spin controller, or type the amount of memory to allocate in the **MB** field. The value must be a multiple of four.

   Note: The minimum size of the memory for the virtual machine should be set based on the recommendations of the operating system provider.

Configuring Virtual Machine Memory from the Management Interface

To set the size of an individual virtual machine's memory from the VMware Management Interface, complete the following steps.

1. Connect to the virtual machine with the management interface.
2. On the Status Monitor page, choose **Configure Hardware**, then click **Edit** next to **Processors and Memory**.


techpubs-linux.vmware.com: winServ2003Std | Configuration - Microsoft Internet Ex...
VMware GSX Server e.x.p build-6624 | root@techpubs-linux.vmware.com
Processors and Memory
Determine processor and memory allocation for your virtual machine.
Edit Processor Configuration
Virtual Machine
Your license only permits virtual machines to be configured with a single processor.
Edit Memory Configuration
Virtual Machine
Suggested: 128 M Guest Min: 128 M Runnable Max: 0 M (1P), 0 M (2P)
Memory (in multiples of 4)
128 M
Help
OK
Cancel

3. Type the amount of memory to allocate to the virtual machine. The value must be a multiple of four.

   Note: The minimum size of the memory for the virtual machine should be set based on the recommendations of the operating system provider.

Improving Performance for Guest Operating Systems

The tips in this section help you make adjustments to improve performance for particular guest operating systems running inside a virtual machine.

The following sections describe tips to improve performance for various guest operating systems:

- Windows 2000, Windows XP and Windows Server 2003 Guest Operating System Performance Tips on page 343
- Windows 95 and Windows 98 Guest Operating System Performance Tips on page 344
- Linux Guest Operating System Performance Tips on page 346

Windows 2000, Windows XP and Windows Server 2003 Guest Operating System Performance Tips

This section offers advice for configuring a Windows 2000, Windows XP or Windows Server 2003 guest operating system for better performance inside a virtual machine.

Note: This section pertains to the guest operating system that is running inside a GSX Server virtual machine. It does not describe actions that should be taken on Windows 2000 or Windows Server 2003 running on the host computer.

Guest Operating System Selection

Make certain you have selected the correct guest operating system in the virtual machine settings editor. Choose **VM** > **Settings** > **Options**.

VMware Tools

Make certain VMware Tools is installed. VMware Tools provides an optimized SVGA driver and sets up the VMware Tools service to run automatically when the system starts. Among other things, VMware Tools allows you to synchronize the virtual machine's clock with the host computer's clock, which can improve performance for some functions. You can install VMware Tools by choosing **VM > Install VMware Tools**.

Disconnect the Virtual CD-ROM Drive

Using the **Removable Devices** submenu, disconnect the virtual CD-ROM drive if you do not need to use it. Disconnecting the CD-ROM drive reduces processor usage.

Visual Effects

The fade effects that Windows 2000, Windows XP and Windows Server 2003 use when displaying menus can be somewhat slow and make the virtual machine seem less responsive.

To disable the fade effects, right-click the guest operating system desktop, then choose **Properties** > **Appearance** > **Effects** (on Windows XP or Windows Server 2003) or **Properties** > **Effects** (on Windows 2000) and deselect the **Use transition effects for menus and tool tips** check box.

Full Screen Mode

Run your virtual machine in full screen mode. Click the **Full Screen** button on the VMware Virtual Machine Console toolbar.

Enabling Hardware Acceleration (Windows Server 2003 Guests Only)

Windows Server 2003 disables hardware acceleration by default. This slows down graphics performance and mouse responsiveness in the guest operating system.

When you install VMware Tools in a Windows Server 2003 guest, you are prompted to enable the hardware acceleration setting. VMware recommends you enable hardware acceleration fully.

To enable hardware acceleration in a Windows Server 2003 guest at a later time, open the Windows Control Panel, then open the Display Properties control panel. On the Settings tab, click **Advanced**. On the Troubleshoot tab, drag the **Hardware acceleration** slider all the way to **Full**.

Windows 95 and Windows 98 Guest Operating System Performance Tips

This section offers advice for configuring a Windows 95 or Windows 98 guest operating system for better performance inside a GSX Server virtual machine.

Guest Operating System Selection

Make certain you have selected the correct guest operating system in the virtual machine settings editor. Choose **VM** > **Settings** > **Options**.

VMware Tools

Make certain VMware Tools is installed. VMware Tools provides an optimized SVGA driver and sets up the VMware Tools service to run automatically when the system starts. Among other things, the VMware Tools service allows you to synchronize the virtual machine's clock with the host computer's clock, which can improve

performance for some functions. You can install VMware Tools by choosing **VM > Install VMware Tools**.

DMA Mode for IDE Disks

Windows 95 OSR2 and higher (including Windows 98) can use direct memory access (DMA) for faster IDE hard disk access. However, DMA may not be enabled by default.

You can turn on DMA access using the guest operating system's Device Manager.

1. Right-click **My Computer** and choose **Properties** from the pop-up menu.
2. Click the **+** sign beside **Disk Drives** to show your virtual machine's individual drives.
3. Right-click the entry for each IDE drive to open its Properties dialog box.
4. Under **Settings**, check the box labeled **DMA** and accept any warning Windows displays.
5. Restart the Windows guest for the new settings to take effect.

Full Screen Mode

Run your virtual machine in full screen mode. Click the **Full Screen** button on the VMware Virtual Machine Console toolbar.

Swap File Usage

In your `system.ini` file, in the `[386enh]` section, add the following line:

```
ConservativeSwapFileUsage=1
```

Disconnect CD-ROM

Using the **Removable Devices** submenu, disconnect your CD-ROM drive if you do not need to use it. Disconnecting the CD-ROM drive reduces processor usage.

Visual Effects

Windows 98 has a number of visual effects, designed to be attractive, that place unnecessary demands on the graphics emulation in GSX Server. Some users have seen performance improvements when they turn off these special effects.

To modify these settings, right-click on the desktop of your virtual machine, then select **Properties** from the pop-up menu. Click the **Effects** tab and deselect the **Animate windows, menus, and lists** check box.

Also, if **Show window contents while dragging** is checked, try deselecting that check box.

Linux Guest Operating System Performance Tips

This section offers advice for configuring a Linux guest operating system for better performance inside a GSX Server virtual machine.

Note: This document pertains to the guest operating system that is running inside a GSX Server virtual machine. It does not describe actions that should be taken on Linux running on the host computer.

Guest Operating System Selection

Make certain you have selected the correct guest operating system in the virtual machine settings editor. Choose **VM** > **Settings** > **Options**.

VMware Tools

Make certain VMware Tools is installed. VMware Tools provides an optimized SVGA driver and sets up the VMware Tools service to run automatically when the system starts. Among other things, the VMware Tools service allows you to synchronize the virtual machine's clock with the host computer's clock, which can improve performance for some functions. You can install VMware Tools by choosing **VM** > **Install VMware Tools**.

Disconnect CD-ROM

Using the **Removable Devices** submenu, disconnect your CD-ROM drive if you do not need to use it. Disconnecting the CD-ROM drive reduces processor usage.

Install in Text Mode

When you are installing your Linux guest operating system, use the text-mode installer instead of the graphical installer if you have a choice. This makes the installation process faster.

If you do use a graphical installer and if you are using a Linux host computer, try to run the virtual machine in full screen mode during the installation.

Full Screen Mode

Run your virtual machine in full screen mode. Click the **Full Screen** button on the VMware Virtual Machine Console toolbar.

CHAPTER 11

Glossary

Add Hardware Wizard — A point-and-click interface for adding virtual hardware to a virtual machine. To launch the wizard, power off the virtual machine, open the virtual machine settings editor, then click **Add**. It prompts you for information for configuring the hardware, suggesting default values in most cases.
See also Virtual machine settings editor.

Bridged networking — A type of network connection between a virtual machine and the rest of the world. Under bridged networking, a virtual machine appears as an additional computer on the same physical Ethernet network as the host.
See also Host-only networking.

Configuration — See Virtual machine configuration file.

Console — See VMware Virtual Machine Console.

Current virtual machine — A virtual machine created under the current version of GSX Server, ESX Server or Workstation.
See also Legacy virtual machine.

Custom networking — Any type of network connection between virtual machines and the host that does not use the default bridged, host-only or network address

translation (NAT) networking configurations. For instance, different virtual machines can be connected to the host by separate networks or connected to each other and not to the host. Any network topology is possible.

EULA — The end user license agreement.

Existing partition — A partition on a physical disk in the host machine.
See also Physical disk.

Full screen mode — A display mode in which the virtual machine's display fills the entire screen.
See also Quick switch mode.

Growable disk — A type of virtual disk where the disk space is not preallocated to its full size. Its files start out small in size and grow as data is written to it.

Guest operating system — An operating system that runs inside a virtual machine.
See also Host operating system.

Headless — A description for a program or application that runs in the background without any interface connected to it. A virtual machine running with no consoles connected to it is considered to be running headless.

Host-only networking — A type of network connection between a virtual machine and the host. Under host-only networking, a virtual machine is connected to the host on a private network, which normally is not visible outside the host. Multiple virtual machines configured with host-only networking on the same host are on the same network.
See also Bridged networking, Custom networking and Network address translation.

Host computer — The physical computer on which the GSX Server software is installed. It hosts the GSX Server virtual machines.

Host operating system — An operating system that runs on the host machine.
See also Guest operating system.

Independent disk — An independent disk is a type of virtual disk that is not affected by snapshots. Independent disks can be configured in persistent and nonpersistent modes.
See also Nonpersistent mode, Persistent mode.

Inventory — A list in the left panel of the console window that shows the names of virtual machines that a user has added to the list. The inventory makes it easy to launch a virtual machine or to connect to the virtual machine's configuration file in order to make changes in the virtual machine settings.

Legacy virtual machine — A virtual machine created under an earlier version of GSX Server, ESX Server or Workstation.
See also Current virtual machine.

Network address translation (NAT) — A type of network connection that allows you to connect your virtual machines to an external network when you have only one IP network address, and that address is used by the host computer. If you use NAT, your virtual machine does not have its own IP address on the external network. Instead, a separate private network is set up on the host computer. Your virtual machine gets an address on that network from the VMware virtual DHCP server. The VMware NAT device passes network data between one or more virtual machines and the external network. It identifies incoming data packets intended for each virtual machine and sends them to the correct destination.
See also Bridged networking, Custom networking and Host-only networking.

New Virtual Machine Wizard — A point-and-click interface for convenient, easy creation of a virtual machine configuration. To launch the wizard, choose File > New Virtual Machine. It prompts you for information, suggesting default values in most cases. It creates files that define the virtual machine, including a virtual machine configuration file and (optionally) a virtual disk or physical disk file.
See also Virtual machine settings editor.

Nonpersistent mode — A mode in which all disk writes issued by software running inside a virtual machine with a disk in nonpersistent mode appear to be written to disk but are in fact discarded after the virtual machine is powered off. If you configure a virtual disk or physical disk as an independent disk in nonpersistent mode, the disk is not modified by GSX Server.
See also Independent disk, Persistent mode

Persistent mode — A mode in which all disk writes issued by software running inside a virtual machine are immediately and permanently written to the virtual disk. If you configure a virtual disk or physical disk as an independent disk in persistent mode, the disk behaves like a conventional disk drive on a physical computer.
See also Independent disk, Nonpersistent mode

Physical disk — A hard disk in a virtual machine that is mapped to a physical disk drive or partition on the host machine. A virtual machine's disk can be stored as a file on the host file system or on a local hard disk. When a virtual machine is configured to use a physical disk, GSX Server directly accesses the local disk or partition as a raw device (not as a file on a file system).
See also Virtual disk.

Preallocated disk — A type of virtual disk where all disk space for the virtual machine is allocated at the time the disk is created. This is the default type of virtual disk created by GSX Server.

Quick switch mode — A display mode in which the virtual machine's display fills most of the screen. In this mode, tabs at the top of the screen allow you to switch quickly from one running virtual machine to another.
See also Full screen mode.

Raw disk — See physical disk.

Redo log — The file that stores the changes made to a disk in independent-nonpersistent mode. The redo-log file is deleted when you power off or reset the virtual machine without writing any changes to the disk.

Resume — Return a virtual machine to operation from its suspended state. When you resume a suspended virtual machine, all applications are in the same state they were when the virtual machine was suspended.
See also Suspend.

Shrink — Reduce the amount of file system space a virtual disk occupies in order to reclaim unused space in a virtual disk. If there is empty space in the disk, shrinking reduces the amount of space the virtual disk occupies on the host drive. Shrinking virtual disks is a convenient way to convert an older virtual disk (created by GSX Server 1, for example) to the `.vmdk` format supported by GSX Server 3. You cannot shrink preallocated virtual disks or physical disks.

Snapshot — A way to preserve the state of a virtual machine — the state of the data on all the virtual machine's disks and the virtual machine's power state (whether the virtual machine was powered on, powered off or suspended). You can take a snapshot of a virtual machine at any time and revert to that snapshot at any time. The virtual machine can be powered on, powered off or suspended.

Supported partition — A virtual disk partition that VMware Tools can prepare for shrinking, such as one of the drives that comprise the virtual hard disk. You can choose to not prepare certain partitions for shrinking.
See also Shrink.

Suspend — Save the current state of a running virtual machine. To return a suspended virtual machine to operation, use the resume feature.
See also Resume.

Unsupported partition — A virtual disk partition that VMware Tools cannot prepare for shrinking. Unsupported partitions include read-only drive partitions, partitions on remote devices and partitions on removable devices such as floppy

drives or CD-ROM drives.
See also Shrink.

Virtual disk — A virtual disk is a file or set of files that appears as a physical disk drive to a guest operating system. These files can be on the host machine or on a remote file system. When you configure a virtual machine with a virtual disk, you can install a new operating system into the disk file without needing to repartition a physical disk or reboot the host. Virtual disks can be preallocated or growable. A preallocated virtual disk has all the disk space allocated at the time the virtual disk is created. A growable disk is not preallocated; its files start out small in size and grow as data is written to it. See also Physical disk.

Virtual hardware — The devices that comprise a virtual machine. The virtual hardware includes the virtual disk, the removable devices such as the DVD-ROM/CD-ROM and floppy drives, and the virtual Ethernet adapter. You configure these devices with the virtual machine settings editor.

Virtual machine — A virtualized x86 PC environment in which a guest operating system and associated application software can run. Multiple virtual machines can operate on the same host machine concurrently.

Virtual machine configuration — The specification of what virtual devices (disks, memory size, and so forth) are present in a virtual machine and how they are mapped to host files and devices.

Virtual machine configuration file — A file containing a virtual machine configuration. It is created when you create the virtual machine. It is used by GSX Server to identify and run a specific virtual machine.

Virtual machine settings editor — A point-and-click control panel used to view and modify a virtual machine's settings. You launch it by choosing VM > Settings. See also New Virtual Machine Wizard.

Virtual Network Editor — A point-and-click editor used to view and modify the networking settings for the virtual networks created by GSX Server. You launch by choosing Host > Virtual Network Settings.

VMware Authorization Service — The service VMware GSX Server employs to authenticate users. The process is called `vmware-authd` on Linux hosts.

VMware Management Interface — A browser-based tool that allows you to control (start, suspend, resume, reset and stop), configure and monitor virtual machines and the server on which they run.

VMware Registration Service — The service VMware GSX Server employs for managing connections to virtual machines and the management interface. This process is known as `vmware-serverd` on Linux hosts.

VMware Tools — A suite of utilities and drivers that enhances the performance and functionality of your guest operating system. Key features of VMware Tools include some or all of the following, depending on your guest operating system: an SVGA driver, a mouse driver, the VMware Tools service, the VMware Tools control panel, and support for such features as the ability to shrink virtual disks, time synchronization with the host, VMware Tools scripts and the ability to connect and disconnect devices while the virtual machine is running.

VMware Tools service — One of the components installed with VMware Tools that performs various duties in the guest operating system, like executing commands in the virtual machine, gracefully shutting down and resetting a virtual machine, sending a heartbeat to VMware GSX Server, synchronizing the time of the guest operating system with the host operating system and passing strings from the host operating system to the guest operating system.

VMware Virtual Machine Console — An interface to a virtual machine that provides access to one or more virtual machines on the local host or a remote host running GSX Server. You can view the virtual machine's display to run programs within it or modify guest operating system settings. In addition, you can change the virtual machine's configuration, install the guest operating system or run the virtual machine in full screen mode.

Index

Symbols

.bmp 105
.cfg 36, 37
.dsk 37
.log 37
.png 105
.REDO 37
.sav 37
.std 37
.vmdk 36, 110, 165
.vmsn 37
.vmss 37
.vmx 36

A

Access to physical disks 200, 220
Adapter
  host virtual 229
  in promiscuous mode on a Linux host 267
  virtual Ethernet 238
Add
  devices to virtual machine 141
  DVD or CD drive 187
  floppy drive 189
  generic SCSI device 326
  host virtual adapter 247
  parallel port 290
  physical disk 184
  serial port 296
  software to virtual machine 139
  virtual disk 177
  virtual Ethernet adapter 238
Add Hardware Wizard 347
Address
  assigning IP 252
  assigning MAC manually 255
  IP in virtual machine 45
  IP on virtual network 250
  MAC 255
  network address translation 269
  using DHCP to assign on a virtual network 250
Assign
  IP address 250
  MAC address 255
Autofit 127
Automatic bridging 242

B

Basic disks on Windows host 213
BIOS
  file in virtual machine 36
  provided in virtual machine 24
Boot loader (LILO) 201, 204, 215
Boot manager 201
Boot sequence in VMware BIOS 203, 205
Bridge 229
Bridged networking
  configuring options 242
Bridged networking defined 347
BSD
  supported guest operating systems 28
  VMware Tools for 71
Build number 81
Buslogic SCSI adapter in virtual machine 46

C

Capture screen shot of virtual machine 105
CD
  adding drive to virtual machine 187
  CD-ROM image file 24
  legacy emulation 115
Change
  See Configure
Chip set in a virtual machine 24
Color screen colors in a virtual machine 334
Commands
  keyboard shortcuts 147
  on the command line 145
Compress
  See Shrink

Configuration, virtual machine 351
Configure
automatic bridging 242
devices in virtual machine 141
DHCP on Linux host 251
DHCP on Windows host 252
DHCP settings 245
display resolution on a Linux host 336
generic SCSI device 325, 326
host virtual network mapping 244
memory size 340
NAT 272
NAT on Linux host 278
networking 242
parallel port 290
parallel port on a Linux host 292
screen colors 334
second bridged network on a Linux host 258
serial port 296
sound 337
USB controller 320
virtual Ethernet adapter 238
virtual network 227, 231, 235, 238
virtual network subnet settings 245
VMware Tools 78
X server 336
Connect
removable devices 144
USB devices 321
Controls, hiding 128
Copy 140
Core files 31
CPU, provided in virtual machine 24
Create virtual machine 35
Current virtual machine 114
defined 347
Cut 140
D
DDNS 258
Defragment virtual disks 168
Delete virtual machine from inventory 104
Devices
adding, configuring and removing 141
connecting and disconnecting 144
disconnecting from USB controller 324
keyboard 310
legacy emulation 115
parallel port 290
provided in virtual machine 24
scanners 325
serial ports 296
tape changers 325
tape drives 325
USB 319
DHCP
assigning IP addresses on a virtual network 250
changing settings 245
configuring on a Linux host 251
configuring on a Windows host 252
DHCP server 230
on a virtual network with NAT 270
server on virtual network 233, 234
troubleshooting on a Linux host 257
dhcpd 257
Dial-up connection 254
Direct memory access
See DMA
Disable host virtual adapter 247
Disconnect
removable devices 144
USB devices 324
Disk files 165
Disk modes
compared to snapshot 158
Disks
adding virtual disks 177
available in virtual machine 24
defragmenting 168
DMA and performance 345
dynamic 213
existing partition 348
file locations 165
growable 163
independent 155, 164
nonpersistent mode 165

Disks *(continued)*
persistent mode 165
physical 349
plain 163
preallocated 162
See also Virtual disk
shrinking 105, 168, 195
size 162
virtual 162, 351
virtual disk files 36
virtual disk manager 191
virtual disk size in new virtual machine 39
Display
color depth 334
fitting virtual machine to window 127
fitting window to virtual machine 127
full screen 125
multiple monitor 126
resolution on a Linux host 336
switching virtual machines 125
DMA and disk performance 225, 345
DNS 270
Drawing tablet in a virtual machine 25
Driver
SCSI 325
sound 337
Drives
See Disks
Dual-boot
and SCSI disks 214
configuring for use in virtual machine 198, 200
Dual-monitor display 126
DVD, adding drive to virtual machine 187
Dynamic disk 213
Dynamic domain name service 258

E

Enable host virtual adapter 247
Ethernet
adapter in a virtual machine 25
adapter in promiscuous mode on a Linux host 267
adding virtual adapter 238
See Network
virtual adapter 230
Event log 31
Existing disk, using in a virtual machine 163
Expand virtual machine screen size 127

F

FAT file system 201
Fault tolerance 242
Files
BIOS in virtual machine 36
location of virtual disk files 39
log 37
redo log 37
renaming virtual disk 110
Samba and file sharing on a Linux host 280
snapshot 37
used by a virtual machine 36
used by snapshot 37
virtual disk files 36
Firewall 276
Fit
virtual machine to window 127
window to virtual machine 127
Floppy
add drive to virtual machine 189
drives in virtual machine 25
image file 25
Floppy disk image file 190
Format
virtual disk 50
FreeBSD
supported guest operating systems 28
VMware Tools for 71
FTP 271
Full screen mode
using 125
Full screen mode, defined 348

G

gated 256

Generic SCSI
adding to virtual machine 326
avoiding concurrent access on a Linux host 331
on a Linux host 330
on a Windows host 328
permissions 331
virtual machine devices 325
Windows NT driver 325
Windows Server 2003 driver 325
Gigabit Ethernet 230, 238
Graphics support in virtual machine 24, 334
Growable disk 163
GSX Server host
powering virtual machines on and off 122
Guest operating system
defined 15, 348
installing 56
supported 27

H

Halt virtual machine 121
Hardware profiles 206
Headless 348
Hide controls 128
Hints 104
Host computer
defined 15, 348
Host operating system, defined 15, 348
Host virtual adapter
adding 247
defined 229
disabling 247
enabling 247
removing 247
Host virtual network mapping 244
Host-only networking
basic configuration 234
defined 348
selecting IP addresses 250

I

ICMP 271
IDE
drives in virtual machine 24
See Disks
virtual disk 47
Image file
floppy 25
ISO 24, 187
Independent disks 164
Install
guest operating system 56
guest operating system on physical disk 220
log 33
silent 69
software in a virtual machine 139
VMware Tools 62
Internet connection sharing 234
Inventory
defined 348
hide 128
removing from list 104
Iomega parallel port Zip drives 295
IP address
assigning 252
in virtual machine 45
IP forwarding 253, 254
ISO image file 24, 187

K

Key code mapping 312
Keyboard
in a virtual machine 25
mapping on a Linux host 310
shortcuts 147
USB 324
Keysym
defined 311
mapping 312
Knowledge base 29

L

Launch virtual machine 107, 110
Leak
IP packets in a virtual machine 254
IP packets in host-only network 253
Legacy emulation 115
Legacy virtual machine 113
defined 349
LILO 201, 204, 215

Linux
keyboard mapping 310
supported guest operating systems 28
VMware Tools for 71
Lock files 167
Log files
console installation 33
GSX Server installation 33
virtual machine 31, 37
virtual machine event log 31
VMware Authorization Service 32
VMware Management Interface 32
VMware Registration Service 32
VMware Virtual Machine Console 31
LSI Logic SCSI adapter in virtual machine 46

M

MAC address 255
Map
key code 312
keyboard 310
keysym 312
Memory
allocating to virtual machine 44
available in virtual machine 24
setting size 340
size for virtual machine 52
virtual machine memory size 340
Migrate, upgrading virtual disks 66
Mode
full screen 125, 348
nonpersistent compared to snapshot 158, 159
persistent compared to snapshot 158, 159
quick switch 125, 350
snapshot and disk modes compared 158
undoable compared to snapshot 158, 159
Mouse
in a virtual machine 25
USB 324
MS-DOS, supported guest operating systems 27

N

NAT
advanced configuration 272
and DHCP 270
and DNS 270
and the host computer 270
defined 349
external access from a NAT network 271
on virtual network 232, 269
sample configuration file for Linux host 278
selecting IP addresses 250
virtual device 230
when creating a virtual machine 45
nat.conf 274, 278
NetWare
See Novell NetWare
Network
adding and modifying virtual Ethernet adapters 238
automatic bridging 242
bridge 229
bridged networking 347
changing DHCP settings 245
changing subnet settings 245
changing the configuration 238
common configurations 231
components 229
configuring 227
configuring bridged networking options 242
custom configurations 235
custom networking 347
DHCP 250
DHCP server 230
dial-up connection 254
dynamic domain name service 258
fault tolerance 242
hardware address 255
host virtual adapter 229
host virtual network mapping 244
host-only 234, 348
host-only subnet 251
Internet connection sharing 234
IP forwarding 253, 254
IP packet leaks 253, 254
MAC address 255

Network *(continued)*
NAT 232, 269
NAT as firewall 276
NAT device 230
NAT subnet 251
packet filtering 254
promiscuous mode on a Linux host 267
routing between two host-only networks 264
routing on a Linux host 256
Samba 280
second bridged network on a Linux host 258
specifying in virtual machine 45
switch 229
teamed NICs 243, 244, 259
Token Ring 233, 234
troubleshooting DHCP on a Linux host 257
two host-only networks 261
using Gigabit Ethernet 230, 238
virtual DHCP server 233, 234
virtual Ethernet adapter 230
Virtual Network Editor 242, 247, 252, 351
virtual switch 229
VirtualCenter 238
VirtualCenter and virtual network adapters 239, 240, 241
wireless network 231
Network adapter in a virtual machine 25
Network adapter teaming 243, 244, 259
Network address translation
See NAT
Network driver
vlance 230, 238
vmxnet 116, 230, 238
New Virtual Machine Wizard 38, 162, 349
NIC
adding and configuring virtual Ethernet adapter 238
promiscuous mode on a Linux host 267
Nonpersistent mode, compared to snapshot 158, 159
Novell NetWare
VMware Tools for 74
Novell NetWare, supported guest operating systems 28
nvram 36, 130

O

Operating system
guest 348
host 348
installing guest 56
supported guest 27

P

Packet
filtering 254
leaks 253, 254
Parallel port
in a virtual machine 25
Parallel ports
and Iomega Zip drives 295
and the Linux kernel 290
configuring on a Linux host 292
in a virtual machine 290
installing in virtual machines 290
Partition
virtual disk 50
Partition, existing 348
Passwords, Samba password file 284
Paste 140
PCI slots
in virtual machine 25
limits 25
Performance
DMA and disks 345
Linux guest 346
memory usage 340
Windows 2000 guest 343
Windows 95 and Windows 98 guests 344
Permissions and running virtual machines 37
Persistent mode, compared to snapshot 158, 159
Physical disk
adding 184
configuring virtual machine on dual-boot computer 198

Physical disk *(continued)*
controlling access 200
defined 349
do not use Windows dynamic disks 213
installing guest operating system on 220
SCSI issues 214
using in a virtual machine 163
using in new virtual machine 39
Ping 271
Plain disk 163
Power buttons for a virtual machine 101
Preallocated disk 162
adding to virtual machine 48, 54
defined 350
Private virtual machines 37
Processor, provided in virtual machine 24
Promiscuous mode 267
PXE boot 132

Q

Quick switch mode 125, 350
Quiet mode, install VMware Tools 69

R

RAM available in virtual machine 24
Raw disk
See Physical disk
Redo log
and snapshots 156
defined 350
Registration 29
Remove
controls 128
devices from virtual machine 141
host virtual adapter 247
removable devices 144
USB devices 324
virtual machine from inventory 104
Rename virtual disk files 110
Repeatable resume
See Resume
Restore
suspended virtual machine 121
virtual machine to state in snapshot 153
Resume
defined 350
snapshot compared to repeatable resume 158
virtual machine 121, 150
Return
See Revert to snapshot
Revert to snapshot 153
routed 256
Routing
between two host-only networks 264
for a host-only network on a Linux host 256
Run suspended virtual machine 121, 150

S

Samba
already running on a Linux host 285
and file sharing on a Linux host 280
and printer sharing 281
on both bridged and host-only networks 285
password file 284
running two Samba servers 286
sample configuration file 281, 282, 286
Save state of virtual machine 121, 129, 150, 153
Scan code 310
Screen colors 334
Screen modes
full screen 125
quick switch 125
Screen shot, capturing 105
Scripts
running during power state changes 76
VMware Tools and 79, 83
SCSI
adding a generic SCSI device 326
and dual-boot configurations 214
avoiding concurrent access on a Linux host 331

SCSI *(continued)*
connecting to generic SCSI device 325
devices in virtual machine 24
disk geometry 217
driver for Windows NT guest 325
driver for Windows Server 2003 guest 214, 325
driver for Windows XP guest 214, 325
drivers 218
generic SCSI on a Linux host 330
generic SCSI on a Windows host 328
permissions for a generic SCSI device on a Linux host 331
virtual disk 47
Serial connection
between host application and virtual machine 300
between two virtual machines 303
to a serial port on the host 296
Serial number 108
Serial port
in a virtual machine 25
installing and using 296
Server
DHCP 230, 237, 245, 251, 257, 270, 277
DNS 258, 270, 273
Samba 280
WINS 271, 277
X on a Linux host 336
Set up
automatic bridging 242
DHCP on Linux host 251
DHCP on Windows host 252
DHCP settings 245
display resolution on a Linux host 336
generic SCSI device 325, 326
host virtual network mapping 244
memory size 340
networking 242
parallel port 290, 296
parallel port on a Linux host 292
screen colors 334
second bridged network on a Linux host 258
software in virtual machine 139
sound 337
USB controller 320
virtual machine 35
virtual network 227, 231, 235, 238
virtual network subnet settings 245
VMware Tools 78
X server 336
Settings for a virtual machine 141
Share files on a Linux host with Samba 280
Shortcuts
keyboard 147
Shrink
defined 350
virtual disks 105, 168, 195
Shut down virtual machine 121
Silent install 69
Size
virtual disk 48
virtual machine window 127
smb.conf 281, 282, 286
SMBIOS
in a virtual machine 24
Snapshot
compared to disk modes 158
compared to repeatable resume 158
defined 350
files 37
redo log 156
removing 157
reverting
using 129
virtual machine 153
ways of using 155
what is saved in 154
Sound
configuring 337
drivers for Windows 9x and Windows NT guest operating systems 337
drivers for Windows Server 2003 guest operating systems 337
in a virtual machine 26
Specifications for virtual machine
platform 24

Start
suspended virtual machine 121, 150
virtual machine 107, 110
VMware Tools 104
Status bar, hide 128
Stop virtual machine 121
Subnet
changing settings 245
in NAT configuration 251
on host-only network 251
Support resources, technical 29
Supported guest operating system 27
Supported partition 350
Suspend
defined 350
virtual machine 121, 150
SVGA
in a Windows 95 guest on a physical disk 210
in a Windows 98 guest on a physical disk 211
Switch, virtual network 229
System Restore, avoiding conflicts 110

T

Tabs, hide 128
Take screen shot of virtual machine 105
Teamed network interface cards 243, 244, 259
Technical support resources 29
Telnet 271
Text, cutting, copying and pasting 140
Token Ring 233, 234
Toolbar
hide 128
power buttons on 101
Tools
installing VMware Tools 62
See VMware Tools
starting VMware Tools 104
Trend Micro Virus Buster
installation issues 139
Turn off virtual machine 121

U

Undoable mode
compared to snapshot 159
Undoable mode, compared to snapshot 158
Uninstall host virtual adapter 247
Unplug USB devices 324
Unsupported partition 350
Upgrade virtual disks 66
USB
connecting devices 321
control of devices by host and guest 322
devices in a virtual machine 319
disconnecting devices 324
enabling and disabling the controller 320
keyboard and mouse 324
on a Linux host 322
on a Windows host 321
supported device types 319
virtual machine ports 25
User groups 29
User interface 97

V

Video
resolution on a Linux host 336
See also Display
Virtual disk
add to virtual machine 177
defined 162, 351
disk type 47
formatting 50
growable 163
location 39
migrating 66
partitioning 50
preallocated 162
See also Disks
shrinking 82, 85, 195, 197
size 39, 48, 178
specifying IDE 47
specifying SCSI 47
Virtual hardware 351

Virtual machine
capturing screen shot of 105
constituent files 36
creating 35
current 114
defined 351
installing software in 139
legacy 113
log 31
moving between VMware products 114
platform specifications 24
power buttons 101
powering off when host shuts down 122
powering on when host starts up 122
private 37
resuming 121, 150
running off CD-ROM/DVD-ROM discs 130
shutting down 121
special power options 118
starting 107, 110
suspending 121, 150
taking and restoring snapshot 129
window size 127
Virtual machine settings editor
defined 351
using 141
Virtual Network Editor 351
Virtual SCSI adapter, BusLogic or LSI Logic 46
Virtual switch 229
VirtualCenter 238
and virtual disk manager 192
and virtual network adapters 239, 240, 241
Virus Buster
See Trend Micro Virus Buster
vlance network driver 230, 238
vmnet1.conf 258
VMnet8 269
VMware Authorization Service
defined 351
log 32
VMware community forums 29
VMware guest operating system service
defined 352
VMware Management Interface
changing virtual machine power state 120
defined 351
log 32
VMware Registration Service
defined 352
log 32
VMware Tools
automated install 69
build number 81
choosing scripts 79, 83
configuring 78
defined 352
for FreeBSD guests 71
for Linux guests 71
for NetWare guests 74
for Windows guests 62
installing 62
running scripts during power state changes 76
silent install 69
starting 104
VMware Tools service 90–94
VMware Tools service
executing commands on halt or reboot 92
passing information between the guest and a VMware Scripting API script 94
passing strings from the host 92
time synchronization 91
VMware Tools 90–94
VMware Virtual Disk Manager 48, 54, 191
VMware Virtual Machine Console 352
connecting 106
connecting from Linux host or client 110
connecting from management interface 111
connecting from Windows host or client 107
log 31
overview 97
vmware.log 37

vmware-authd
See VMware Authorization Service
vmware-config-tools.pl 73
vmware-serverd
See VMware Registration Service
vmxnet network driver 230
and older virtual machines 116
configuring 238
virtual network component 230
V-scan code
defined 311
table of codes 314
W
Window size 127
Windows
VMware Tools for 62
Windows 95
sound driver 337
SVGA driver in a physical disk configuration 210
Windows 98
sound driver 337
SVGA driver in a physical disk configuration 211
Windows NT
SCSI driver for guest 325
sound driver 337
Windows Server 2003
installing guest operating system 56
SCSI driver for guest 214, 325
sound driver 337
Windows XP, SCSI driver for guest 214, 325
Windows, supported guest operating systems 27
Wireless network 231
Wizard
add hardware 347
new virtual machine 38, 165, 349
X
X server
and keyboard mapping 310
configuring 336
XFree86
and keyboard mapping 310
configuring 336
Z
Zip drives on a parallel port 295